Journal of Electrical and Computer Engineering

Resource Allocation in Communications and Computing

Guest Editors: Yi Su, Fangwen Fu, and Shuo Guo

Resource Allocation in Communications and Computing

Journal of Electrical and Computer Engineering

Resource Allocation in Communications and Computing

Guest Editors: Yi Su, Fangwen Fu, and Shuo Guo

Editorial Board

The editorial board of the journal is organized into sections that correspond to the subject areas covered by the journal.

Circuits and Systems

Communications

Signal Processing

Contents

Resource Allocation in Communications and Computing, Yi Su, Fangwen Fu, and Shuo Guo
Volume 2013, Article ID 328395, 2 pages

A Rendezvous Protocol with the Heterogeneous Spectrum Availability Analysis for Cognitive Radio Ad Hoc Networks, Sylwia Romaszko
Volume 2013, Article ID 715816, 18 pages

Multidomain Hierarchical Resource Allocation for Grid Applications,
Mohamed Abouelela and Mohamed El-Darieby
Volume 2012, Article ID 415182, 8 pages

Virtual Network Embedding: A Hybrid Vertex Mapping Solution for Dynamic Resource Allocation,
Adil Razzaq, Markus Hidell, and Peter Sjödin
Volume 2012, Article ID 358647, 17 pages

Bit Rate Optimization with MMSE Detector for Multicast LP-OFDM Systems, Ali Maiga,
Jean-Yves Baudais, and Jean-François Hélard
Volume 2012, Article ID 232797, 12 pages

Analytical Evaluation of the Performance of Proportional Fair Scheduling in OFDMA-Based Wireless Systems, Mohamed H. Ahmed, Octavia A. Dobre, and Rabie K. Almatarneh
Volume 2012, Article ID 680318, 12 pages

Optimizing Spectrum Trading in Cognitive Mesh Network Using Machine Learning,
Ayoub Alsarhan and Anjali Agarwal
Volume 2012, Article ID 562615, 12 pages

Comparison Study of Resource Allocation Strategies for OFDM Multimedia Networks,
Cédric Guéguen and Sébastien Baey
Volume 2012, Article ID 781520, 11 pages

Resource Management in Satellite Communication Systems: Heuristic Schemes and Algorithms,
Shahaf I. Wayer and Arie Reichman
Volume 2012, Article ID 169026, 10 pages

Robustness Maximization of Parallel Multichannel Systems, Jean-Yves Baudais, Fahad Syed Muhammad, and Jean-François Hélard
Volume 2012, Article ID 840513, 16 pages

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2013, Article ID 328395, 2 pages
http://dx.doi.org/10.1155/2013/328395

Editorial
Resource Allocation in Communications and Computing

Yi Su,[1] Fangwen Fu,[2] and Shuo Guo[3]

[1] *Qualcomm Incorporated, Santa Clara, CA 95051, USA*
[2] *Intel Incorporated, Folsom, CA 95630, USA*
[3] *University of Minnesota at Twin Cities, Minneapolis, MN 55455-0213, USA*

Correspondence should be addressed to Yi Su; yis@qualcomm.com

Received 29 August 2012; Accepted 29 August 2012

Communication networks and computing systems have demonstrated their importance in the past few decades as a fundamental driver of economic growth. Over the years, they have not only expanded in their sizes, such as geographical area and number of terminals, but also in the variety of services, users, and deployment environments. The purpose of resource allocation in such environments is to intelligently assign the limited available resources among terminals/clients in an efficient way to satisfy end users' service requirements.

With the dramatic developments and fast evolution of communication networks and computing systems, resource allocation continues to be the fundamental challenge, because better quality of service is required with the increasing demand for bandwidth-hungry and/or computation-intensive services. In particular, it has to cope with various new emerging system architectures, such as cognitive networks, mesh networks, multihop networks, peer-to-peer networks, multistandard networks, cloud computing systems, and data centers, distributed intelligence in a multitude of devices operating autonomously enables shifting traditional centralized allocation mechanisms into fully distributed solutions. In recent years, many tools including optimization theory, control theory, game theory, and auction theory have been employed to model and solve a variety of practical resource allocation problems. Therefore, resource allocation in communication networks and computing systems is a pressing research topic that has huge applications. It is imperative to develop advanced resource allocation techniques for ensuring the optimal performance of these systems and networks.

The goal of this special issue is to bring together the most updated research contributions in this area. Indeed, we see a wide range of new analytical techniques and novel application scenarios emerging as evidenced in the papers presented here. The nine accepted papers are relevant to resource allocations optimizations in orthogonal frequency division multiplexing (OFDM-) based communication systems, cognitive radio, satellite communications, grid computing, and network virtualization.

J. Y. Baudais et al. in "*Robustness maximization of parallel multichannel systems,*" study bit-loading solutions of both robustness optimization problems over independent parallel channels. Their investigation is based on analytical approach, using generalized Lagrangian relaxation tool, and on greedy-type algorithm approach. The asymptotic convergence of both robustness optimizations is proved for both analytical and algorithmic approaches. They also link the SNR-gap maximization problem to the conventional power minimization problem and prove that the duality does not hold in all cases. In nonasymptotic regime, they show that the resource allocation policies can be interchanged depending on the robustness measure and the operating point of the communication system. They propose a low-complexity resource allocation algorithm based on the analytical approach, which leads to a good tradeoff between performance and complexity.

C. Guéguen and S. Baey, in "*Comparison study of resource allocation strategies for OFDM multimedia networks,*" present and compare the main OFDM scheduling techniques used for multimedia services in multiuser OFDM wireless networks. They study the influence of bandwidth granularity on the resource allocation strategies performances. They show that

bandwidth granularity is of major importance for determining the application range of advanced OFDM scheduling techniques.

M. H. Ahmed et al. in "*Analytical evaluation of the performance of proportional fair scheduling in OFDMA-based wireless systems,*" evaluate the performance of proportional fair (PF) scheduling in orthogonal frequency division multiple access (OFDMA) wireless systems. They investigate a two-dimensional (time slot and frequency subcarrier) PF scheduling algorithm for OFDMA systems and evaluate its performance analytically and by simulations. They derive approximate closed-form expressions for the average throughput, throughput fairness index, and packet delay. Computer simulations show good accuracy of the analytical expressions.

A. Maiga et al. in "*Bitrate optimization with MMSE detector for multicast LP-OFDM system,*" propose a new resource allocation algorithm with minimum mean square error (MMSE) detector for multicast linear precoded (LP) OFDM systems. They propose to jointly use the LP-OFDM modulation technique and an adaptation of the OFDM-based multicast approaches to exploit the transmission link diversities of users and improve both the bit rate and the fairness among multicast users.

S. Romaszko and P. Mahonen, in "*A rendezvous protocol with the heterogeneous spectrum availability analysis for cognitive radio ad hoc networks,*" look into a new challenge problem of rendezvous (RDV) protocol in cognitive radio ad hoc networks (CRANs). In such a frequently changing environment, licensed holders channel occupancy, and heterogeneous spectrum availability result in a need of on-demand searching for a control traffic channel by CR users in order to be able to initiate a communication and methods guaranteeing that all nodes meet periodically in reasonable periods of time should be advocated. They evaluate a torus quorum system (QS) and difference set (DS) based rendezvous protocol (MtQS-DSrdv) and show that the nodes meet multiple times on different channels in a period, which increases the chance of successful establishment of a real communication.

A. Alsarhan and A. Agarwal, in "*Optimizing spectrum trading in cognitive mesh network using machine learning,*" propose a reinforcement learning (RL) model in a cognitive wireless mesh network for licensed users (primary users, PUs) to maximize the revenue of renting surplus spectrum to unlicensed users (secondary users, SUs). They use RL extract the optimal control policy that maximizes the PUs' profit continuously over time. The extracted policy is used by PUs to manage renting the spectrum to SUs, and it helps PUs to adapt to the changing network conditions. They also propose a new distributed algorithm to manage spectrum sharing among PUs to maximize the total revenue and utilize spectrum efficiently.

S. Wayer and I. Reichman, in "*Resource management in satellite communication systems-heuristic schemes and algorithms,*" study the challenging resource allocation problem in satellite communication due to the high cost of frequency bandwidth. They define a satisfaction measure to estimate the allocation processes and carry out resource management according to the requests of subscribers, their priority levels, and assured bandwidths.

M. Abouelela and M. El-Darieby, in "*Multi-domain hierarchical resource allocation for grid applications,*" propose a hierarchical-based architecture as well as multidomain hierarchical resource allocation approach for geographically distributed applications in grid computing environments. They perform the resource allocation in a distributed way among different domains such that each participant domain keeps its internal topology and private data hidden while sharing abstracted information with other domains. The proposed algorithm jointly schedules computing and networking resources while optimizing the application completion time taking into account data transfer delays.

A. Razzaq et al. in "*Virtual network embedding: a hybrid vertex mapping solution for dynamic resource allocation,*" investigate the problem of virtual network embedding (VNE) in the context of network virtualization. They analyze two existing vertex mapping approaches and propose a new vertex mapping approach which minimizes complete exhaustion of substrate nodes while still providing good overall resource utilization. They also investigate under which circumstances the proposed vertex mapping approach can provide superior VN embedding properties.

Before closing this editorial, we would like to thank those who contributed significantly behind the scene towards the success of this special issue. We hope that you will enjoy reading this Special Issue devoted to the exciting fast-evolving field of resource allocation in communications and computing as much as we have done.

Yi Su
Fangwen Fu
Shuo Guo

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2013, Article ID 715816, 18 pages
http://dx.doi.org/10.1155/2013/715816

Research Article

A Rendezvous Protocol with the Heterogeneous Spectrum Availability Analysis for Cognitive Radio Ad Hoc Networks

Sylwia Romaszko

Institute for Networked Systems, RWTH Aachen University, Kackertstraße 9, 52072 Aachen, Germany

Correspondence should be addressed to Sylwia Romaszko; sar@inets.rwth-aachen.de

Received 4 May 2012; Revised 11 August 2012; Accepted 20 August 2012

Academic Editor: Fangwen Fu

In cognitive radio ad hoc networks, a frequently changing environment, varying channel occupancy statistics, and heterogeneous spectrum availability result in a need to meet on a common channel and to initiate a communication. This process of two or more CRs, meeting each other in the same channel, is called a rendezvous (RDV). RDV is essential for establishment of a communication link. Hence, methods guaranteeing that all nodes meet periodically in reasonable periods of time should be developed. In this study, we evaluate a torus Quorum System (QS) and Difference Set (DS) based rendezvous protocol in an asymmetric channel view case (heterogeneous channel availability). Regardless of the diversity of channels of CRs the protocol guarantees RDV on either all channels or almost all channels. Furthermore, the nodes meet multiple times on different channels in a period, which increases the chance of successful establishment of a communication link.

1. Introduction

Cognitive Radio Networks (CRNs) can operate in licensed and unlicensed bands. A spectrum hole is a frequency band that has been assigned to a primary user (PU) but is not utilized by this PU at a particular time and specific geographic location [1]. Secondary Users (SUs), also called Cognitive Radio (CR) users, have only an opportunistic spectrum access to these bands, that is, the licensed spectrum; that is, temporarily vacant may be used by a CR user [2]. The appearance of a PU means that CRs must vacate immediately the occupied frequency band. Hence, link recovery information (and a new determined channel) cannot be circulated over the previously used spectrum band. The dissemination of control traffic signals (on a common control channel, CCC) must be fast, since the SU might have a limited duration of time in which a spectrum hole is likely to be available. Furthermore, such dissemination should be robust and minimize the use of energy and computing resources [1]. Hence, the classical CCC of multichannel networks is not an attractive solution for CRNs. Moreover, unlike in the case of classical ad hoc networks, a CR has the heterogeneous spectrum availability which is varying over time and space due to the licensed holders' activities. In other words, the available radio resources (channel set) can be different for different CRs (also known as asymmetric channel occupancy knowledge) in the same network due to different location or PUs activities. From all these unique CRN characteristics arises a big research challenge; namely, how to achieve a rendezvous between nodes. Haykin's question, *how can we establish the dissemination of control traffic signals between neighboring SUs in cognitive radio ad hoc networks, which is rapid, robust, and efficient* [1], is still not fully answered and is a challenging problem.

In this study, we investigate a distributed rendezvous protocol for cognitive radio ad hoc networks employing frequency hopping (FH) techniques in the case of heterogeneous channel availability. FH is known for decreasing the probability of interference to PUs thanks to the frequent switching of the occupied channels. However, some undesirable assumptions of typical FH (e.g., the need of synchronization and exchange of hopping patterns, the same length of hopping sequence patterns) must be overcome (e.g., incumbents presence and as consequence the need to vacate the channel immediately by a SU; heterogeneous channel availability, etc.). Before going into details of the protocol, we formulate the rendezvous (RDV) problem with regard to channel switching and an asymmetric channel occupancy

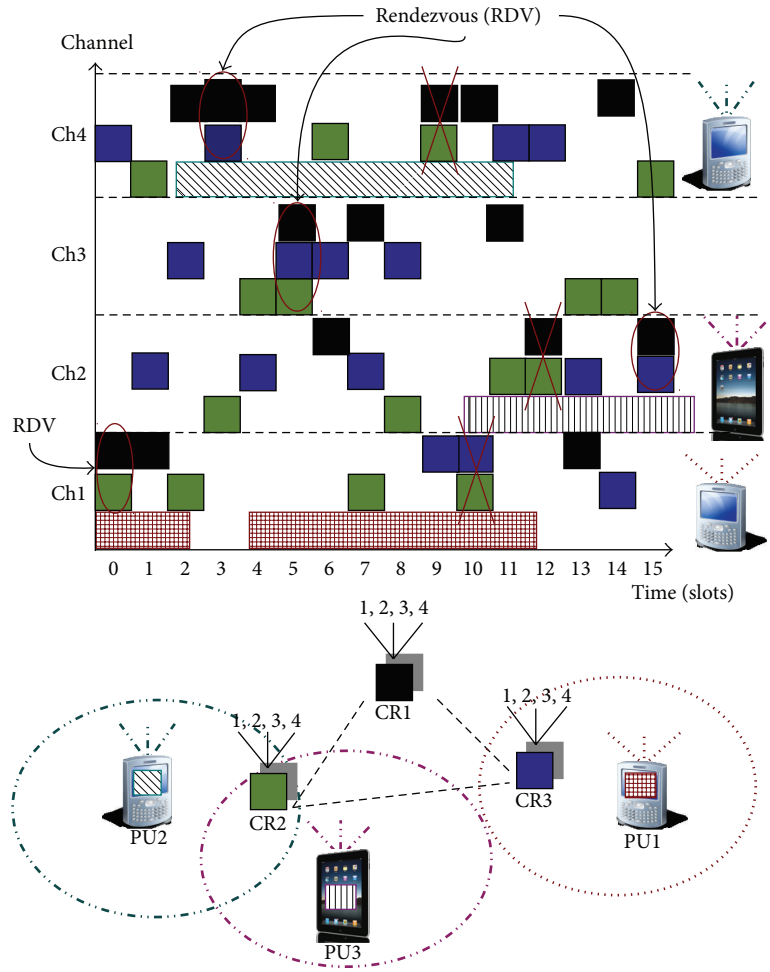


Figure 1: Rendezvous in CRNs: if there is no hatched rectangle of a PU on a particular channel, it means that this PU is not active on this channel; for example, there is no PU's activity on Channel 3 (Ch3). CR1, not being in the range of any of PUs, can use all four channels. CR3 can sporadically use Channel 1 due to the PU1 activity there. CR2 has two available channels (1, 3), while the availability in Channels 2 and 4 is limited to spectrum holes of PU2 and PU3. Hence, CRs have an *asymmetric channel occupancy knowledge*. In a 16 slots of long time period, CR1 user can meet CR2 user on Channel 1 in slot 0 and on Channel 3 in slot 5. However, they cannot communicate on Channel 4 in slot 9, nor on Channel 2 in slot 12, due to the PU2 and PU3 activities there (CR2 cannot transmit).

knowledge. A RDV problem in multichannel networks refers to the ability of two or more CRs to meet each other in the same channel.

Figure 1 illustrates an example of the channel occupancy by three primary users (hatched rectangles) and three secondary users (solid filled squares or rectangles) hopping through four channels (Ch1...Ch4) in a 16 slots of long cycle (for the sake of simplicity we illustrate the case where secondary users have a common time-slot system. This is done without losing generality, and, in fact, we have also presented asynchronous operations in [3]). Here, by a slot we understand some period of time within which SUs can communicate with each other (i.e., exchanging information, receiving and transmitting). The situation depicted in Figure 1 is very optimistic; nodes are able to meet once, twice or even three times (CR1 and CR3). However, while hopping or switching channels randomly it might happen that nodes never meet. In order to deal with the aforementioned RDV problem, we proposed in [4] the protocol which is based on the torus quorum system (tQS) concept.

Thanks to the use of the quorum system (QS) and also the difference set (DS) concept we are able to guarantee meeting on all r channels in the case of nodes with a symmetric channel view (SCHv), and almost on all channels in an asymmetric channel view (ACHv) case. We emphasize that CR nodes do not need to have mutual knowledge of their hopping sequences information because the use of QS properties is sufficient. Thanks to the QS intersection property guarantee cognitive radios will meet (intersect) when selecting quorum-based hopping sequences from the same QS. Quorums, satisfying the Rotation Closure Property, guarantee the intersection even if the cycles of the CRs are not aligned and, therefore, can be used in asynchronous protocols. The torus QS [5], exploited in this work, represents an intuitive and simple method to employ where a quorum set is chosen by selecting a column of an $r \times s$ rectangular

array (r is the number of rows and s is the number of columns) and $\lfloor s/2 \rfloor$ elements from other columns. Other QS methods are much more complex, for example, cyclic QS (a cyclic QS is based on the cyclic block design and cyclic DSs in combinatorial theory [6]) proposed in [7]. The QS concept itself has already been used [8] in the context of operating systems. In the last decade, the use of QSs has been extended to other applications, for example, power-saving protocols (e.g., [9, 10]). Quorum systems are also adopted in order to overcome a rendezvous problem in multichannel networks (e.g., [11–14]). For additional review in the context of cognitive radios we refer the reader to [4].

The content of this work is as follows. For sake of understanding the analyzed algorithm, we give a short introduction to quorum systems focusing on a torus QS and Difference Sets. We describe the standard method of constructing a torus quorum [5] (forward manner) and the mirror method [15]. Afterwards, we shortly describe the construction rules of the MtQS-DSrdv protocol designed in our previous work [4], followed by a throughout performance evaluation. In a recent survey paper [4] we also discussed a possibility to use MtQS-DSrdv for RDV in the case of cognitive radio networks. However, that discussion was more of illustration of a possibility and contained only preliminary results. In this paper we will provide more in-depth analysis and consider also asymmetric channels. We provide additional analysis in terms of time-to-rendezvous (TTR) and the mean occurrence of rendezvous on a channel within a hopping sequence/period. In contrast to [4], we also consider the *asymmetric* channel view in terms of (maximal) (M)TTR, RDV on x channels, and RDV occurrence on a channel within a hopping sequence/period. Furthermore, as opposed to [4], we show the advantage of the MtQS-DSrdv protocol against the related work both in SCHv and/or ACHv in terms of (M)TTR metric and the probability of RDVs on different channels in one hopping sequence period.

Section 2 presents the related work in this area. In Section 3 we describe the QS and DS concepts with all relevant definitions and properties. The system model is presented in Section 4. We prompt the MtQS-DSrdv protocol in Section 5. The extensive evaluation of the protocol in the symmetric and asymmetric channel view is described in Section 6, with the comparison with the selected related work in the last subsection. Section 7 concludes this study.

2. Related Work

A strict coordination or some degree of synchronization between nodes is often assumed in dynamic spectrum access literature. Either a TDMA or FH-like access schemes are used with the assumption that nodes can synchronize or coordinate easily; see, for example, [16–19]. In [20] taxonomy, challenges, classification, and comparison of rendezvous approaches in CRNs can be found.

Here we summarize the most recent related work by categorizing methods into three branches. The first category comprises nonquorum-based solutions representing blind or pseudorandom RDV techniques [21–26]. The second branch contains either QS-based protocols proposed for a multichannel Medium Access Control (MAC) [11, 27, 28] or non-QS but sequence-based algorithms for CRNs [29–31]. Finally in the third branch we have a number of quorum system-based protocols proposed for CRNs [12–14, 32–34].

A-MOCH [30], the non-QS-based protocol for CRNs, is based on Latin Square (LS) (transmitter) and Identical-Row Square (IRS) maps (receiver). This approach guarantees RDV on all channels in a period, but only once on each channel. Moreover, while being in the reception mode receivers need to switch channels constantly, which is definitely not desirable for wireless resource-constrained systems. It might also happen (cf. [30]) that a transmitter will select such LS, which also implies switching channels constantly. In other words, in this approach the cost of channel switching time for a receiver should be taken into account.

In the asynchronous ASYNC-ETCH algorithm, presented in [31], there is no need of global clock synchronization, no matter how the hopping processes of nodes are misaligned. A hopping sequence S_i is composed of N frames (N denotes the number of available channels), where each frame is composed of a pilot slot and two subsequences sub S_i. The pilot slots of S_i, collected together, are the channels appearing in sub S_i in the same order. The hopping sequence is equal to $(2N + 1) \times N$; for example, with 5 channels, it is composed of 55 slots. If the pair of CRs selects the same hopping sequence, RDV is guaranteed in one slot per hopping period. However, while using different sequences, there is RDV guaranteed in N slots.

In order to have a RDV channel without the help of CCC or synchronization, and to guarantee rendezvous in at least one channel for each searching sequence, the Balanced Incomplete Block Design (BIBD) [35] has been used in [29]. Authors introduce single-sequence and multisequence MAC protocols, where the latter builds hopping sequences for the multichannel case. The protocol is compared to the permutation sequence proposed in [36] and blind rendezvous in [37].

In DSMMAC [13] all nodes create the same channel hopping sequence, and the only possible variation of this sequence is dependent on an offset (if cycles are not aligned). Moreover, the process of the forming of channel hopping sequences is not easy; namely, it is based on the Difference Sets which must be chosen in a very careful manner in order to ensure a high RDV probability.

Since a torus QS is a special case of a grid QS, we also consider two grid quorum-based rendezvous algorithms [14, 34]. The schemes do not guarantee RDV, although the percentage of missed RDV is very low.

In [4] we gave a comprehensive guidance on the use of quorum systems. We also addressed RDV issues in decentralized CRNs surveying exhaustively channel hopping approaches. Additionally, channel hopping requirements for cognitive radio ad hoc networks have been proposed, and the most suitable related work to the RDV problem has been appraised according to those requirements. The MtQS-DSrdv protocol rules were proposed in [4] along with its assessment according to the proposed requirements.

In [33] a sequence-based protocol has been implemented on Universal Software Radio Peripheral (USRP) boards and

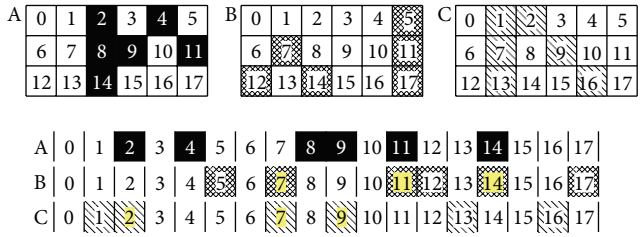


Figure 2: Standard torus QS example: node A is built by picking the third column as its head and 3 randomly chosen slots from succeeding columns. B is formed by selecting the sixth column and its tail from succeeding columns in a wrap-around manner. C's head is the second column. A and B intersect at slots 11 and 14, B and C at slot 7, and A and C at slots 2 and 9.

evaluated in terms of the time of the first encounter between two SUs and the time for encounter on all channels. The protocol itself is partly based on QS [38, 39]. The experiments show a big advantage of the sequence-based scheme over the random sequence scheme, with and without PU's presence.

In contrary to gQ-RDV method [34] or BIBD-based approach [29], the MtQS-DSrdv protocol guarantees rendezvous on all r available channels in a symmetric channel view. In comparison to A-MOCH [30] our scheme does not use two different sequences for receiver and transmitter, nodes switch channels less frequently, and nodes can meet more than once on the same channel during one period.

3. Torus Quorum System and Difference Set Properties

In this section we present the (torus) QS and DS concepts. Since QSs are not commonly used in wireless communications, we present first some relevant definitions in this section.

A torus-based QS (tQS) [5] adopts a rectangular array structure called *torus*, that is, wrap-around mesh, where the last row (column) is followed by the first row (column) in a wrap-around manner (later in this work, we call the standard tQS as the *forward* tQS, since its tail is selected going forward (to the next columns)). The height r (number of rows, i.e., entire column) and width s (number of columns, i.e., entire row) are defined where $n = r \times s$ and $s \geq r \geq 1$.

Definition 1 (Torus Quorum Systems). A torus quorum in a $r \times s$ torus is composed of $r + \lfloor s/2 \rfloor$ elements, formed by selecting any column C_j ($j = 1 \cdots s$) of r elements plus one element out of each of the $\lfloor s/2 \rfloor$ succeeding columns using end wrap-around. An entire column C_j portion is called the quorum's *head*, and the rest of the elements ($\lfloor s/2 \rfloor$) its *tail*.

Figure 2 gives an example for three nodes choosing three different torus quorums under $U = \{0, 1, \ldots, 17\}$ where $r = 3$ and $s = 6$, thus $n = 18$.

It is also possible to construct a tQ in a backward manner as shown in [40]. However, in order to select a torus tail in a more flexible manner, the *mirror* torus extension should be used [4, 15], which allows to alternate selecting tail's slot in a forward or backward manner.

Definition 2 (Mirror Torus Extension). A tail of a torus quorum, $\lfloor s/2 \rfloor$ elements, can be selected from any position of column C_{j+k_i*i} (one element from a column), where $k_i \in \{1, -1\}$ and $i = 1 \cdots \lfloor s/2 \rfloor$, in a wrap-around manner. Toruses of the same torus QS need to select elements in the same forward/backward order.

In other words, if an element was selected from column C_{j+1}, the next element cannot be selected from C_{j-1}, but needs to originate from the next succeeding (forward) column (C_{j+2}) or the preceding (backward) column (C_{j-2}). The parameter k_i needs to be the same for all quorums of the same torus QS; that is, the direction of the selection needs to be the same. Figure 3 shows the selection in a mirror way.

The intersection property of quorum systems is not sufficient when the cycle of nodes is not aligned or nodes are asynchronous. In order to have RDV guarantee in such case, a quorum must satisfy the Rotation Closure Property (RCP).

Definition 3 (Rotation Closure Property). For a quorum R in a quorum system Q under an universal set $U = \{0, \ldots, n-1\}$ and $i \in \{1, 2, \ldots, n-1\}$, one defines: rotate$(R, i) = (x + i) \bmod n | x \in R$. A quorum system Q has the Rotation Closure Property if and only if

$$\forall R', \quad R \in Q, R' \cap \text{rotate}(R, i) \neq \emptyset \quad \forall i \in 1, 2, \ldots, n-1. \tag{1}$$

A quorum system, which satisfies the Rotation Closure Property, ensures that two asynchronous mobile nodes selecting any two quorums have at least one intersection in their quorums. The (forward, backward, and mirror) torus quorum satisfies the Rotation Closure Property.

The Difference Sets [6] concept is very close to QSs, being actually the basis of the cyclic QS.

Definition 4 (Cyclic Difference Set (DS)). A subset B, such as $B = \{a_1, a_2, \ldots, a_k\}$modulo n, for $a_i \in 1, 2, \ldots, n-1$, is called a cyclic (n, k, λ) difference set under Z_n (k and λ are positive integers such that $2 \leq k < n$ and $|B| = k$), if for every $b \not\equiv 0 (\bmod n)$ there are exactly λ ordered pairs (a_i, a_j), where a_i, $a_j \in B$ in such a way that $a_i - a_j \equiv b (\bmod n)$.

If at least one ordered pair (a_i, a_j) exists in (n, k) difference set, such set is called a relaxed DS.

Since tQs also form DSs we show an example based on the set from Figure 2. Node A has a set $\{2, 4, 8, 9, 11, 14\}$ where $n = 18$. The set is a relaxed DS, because there is at least one ordered pair (a_i, a_j):

$$\begin{gathered} 1 \equiv 9 - 8, \quad 2 \equiv 4 - 2, \quad 3 \equiv 11 - 8, \quad 4 \equiv 8 - 4, \\ 5 \equiv 9 - 4, \quad 6 \equiv 8 - 2, \quad 7 \equiv 9 - 2, \quad 8 \equiv 4 - 14, \\ 9 \equiv 11 - 2, \quad 10 \equiv 14 - 4, \quad 11 \equiv 2 - 9, \\ 12 \equiv 2 - 8, \quad 13 \equiv 9 - 14, \quad 14 \equiv 4 - 8, \\ 15 \equiv 11 - 14, \quad 16 \equiv 2 - 4, \quad 17 \equiv 8 - 9. \end{gathered} \pmod{18}$$

The reader should note that each cyclic DS satisfies the RCP.

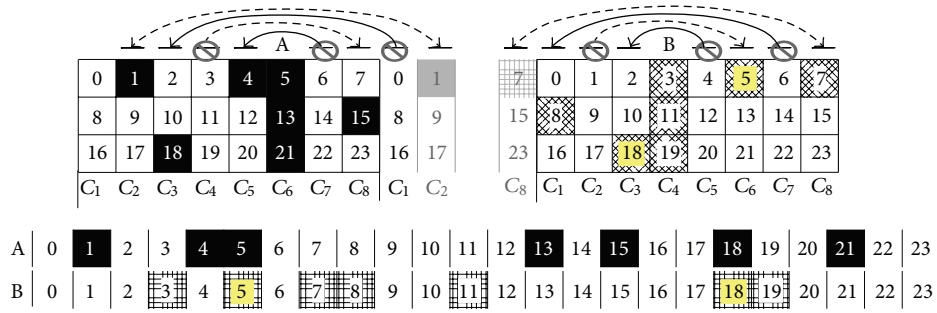


Figure 3: Example of mirror torus QS: A and B initiate their tail selection in a backward manner; that is, the first element of node A is selected from column C_5 (a mirror of C_7), and the one of node B is selected from column C_3 (a mirror of C_5). Then, both nodes select the next tail element in a forward manner, thus from C_{j+2} column, which is C_8 in case of node A, and C_6 in case of node B. The third tail element is selected in a backward manner, which is C_3 in case of node A, and C_1 in case of node B. The last element of nodes A and B falls in the same (mirror) column. Nodes meet at slot 5 (they would meet in one element of column C_6 anyway) and additionally in slot 18. Hence, nodes can always alternate the manner, either forward or backward, while picking up elements; however, they need to have the same k_i, in this example $k_i = \{-1, 1, -1, \{1 \mid -1\}\}$, where $i = 1 \cdots \lfloor 8/2 \rfloor$.

4. System Model

We focus on Secondary Users in the presence of Primary Users in the network, where no central units for management of spectrum allocation are present. The rendezvous on the same channel (hereafter terms channel and frequency will be used interchangeably) between two SUs is crucial for the establishment of the communication. Each SU is equipped with a single tunable half-duplex radio transceiver which can switch between r different channels. In cognitive radio networks SUs must somehow identify spectrum holes, which vary in time and space, and after that select available frequencies. Based on a spectrum detection method (sensing, database) each CR recognizes a list of spectrum holes that can be used while respecting PU priority. It is assumed that channels are slowly time-varying, and that the system is slowly dynamic.

SUs should find each other periodically and as soon as possible (i.e., Time-to-Rendezvous should be small and bounded (TTR); is an amount of time, measured in slots, within which two or more CRs meet each other once they began hopping, or after the last RDV on a channel). Here, we emphasize that the need of rendezvous on *multiple* available channels in a hopping sequence period is paramount, as thanks to the guarantee of this property, the channel access delay will be minimized. If an RDV protocol cannot satisfy a periodic overlap between channel hopping sequences of cognitive radios on different channels, this raises an RDV problem, because a *single* rendezvous channel might become unavailable due to the (sudden) appearance of primary user signals. If an RDV protocol can guarantee a rendezvous on *every* available channel, it provides the maximum robustness. Therefore, we aim to guarantee rendezvous on every available channel in the case of the homogeneous spectrum availability (i.e., all SUs have the same channel set), and on multiple different available common channels in the case of the heterogeneous spectrum availability. The latter objective is challenging, not only due to the different channels sets of CRs, but also because of different lengths of channel hopping sequences.

In our previous work [4] we developed MtQS-DSrdv algorithm and showed that in the case of a symmetric channel view (homogeneous spectrum availability), CR nodes meet *periodically* on every available channel. As explained before, a periodic overlap is guaranteed thanks to the use of QS properties, and therefore there is no need of exchange of any information in order to meet in a hopping sequence period. In this work we extend the analysis of SCHv in terms of TTR and we evaluate the protocol for an asymmetric channel view (heterogeneous spectrum availability).

In the MtQS-DSrdv algorithm we consider that each SU hops from one channel to another (Figure 1) according to its frequency hopping sequence in order to discover another SU. Each CR determines its channel map for each of the channels making use of a torus array ($r \times s$) using torus QS and DS concepts and then combines them into a hopping sequence. This process requires no mutual knowledge of hopping sequence information and available channels from other CRs. The resulting hopping sequence is cyclic and it counts as many slots as there are elements in the torus array. Every element represents a time slot where a single channel is designated to be used. Here, we stress that a slot is an amount of time within which cognitive radio users can communicate, that is, discover each other by exchanging messages.

5. MtQS-DSrdv Epitome

In this section we describe MtQS-DSrdv algorithm. While forming the channel map (hopping sequence), two concepts are employed, namely, torus Quorum System and Difference Sets. The former is straightforward, since we just select a torus quorum algorithm while the head (column) should follow the construction rules of the algorithm, and tail can be chosen randomly. The reader should note that with $r > 4$ the first four channels have torus Quorum System-based maps, and the rest have Difference Set-based maps.

Figure 4 depicts how each node constructs its hopping sequence with a cycle of n slots. The column selection of Channel 1 (step 2.a.1) specifies the start point of a map

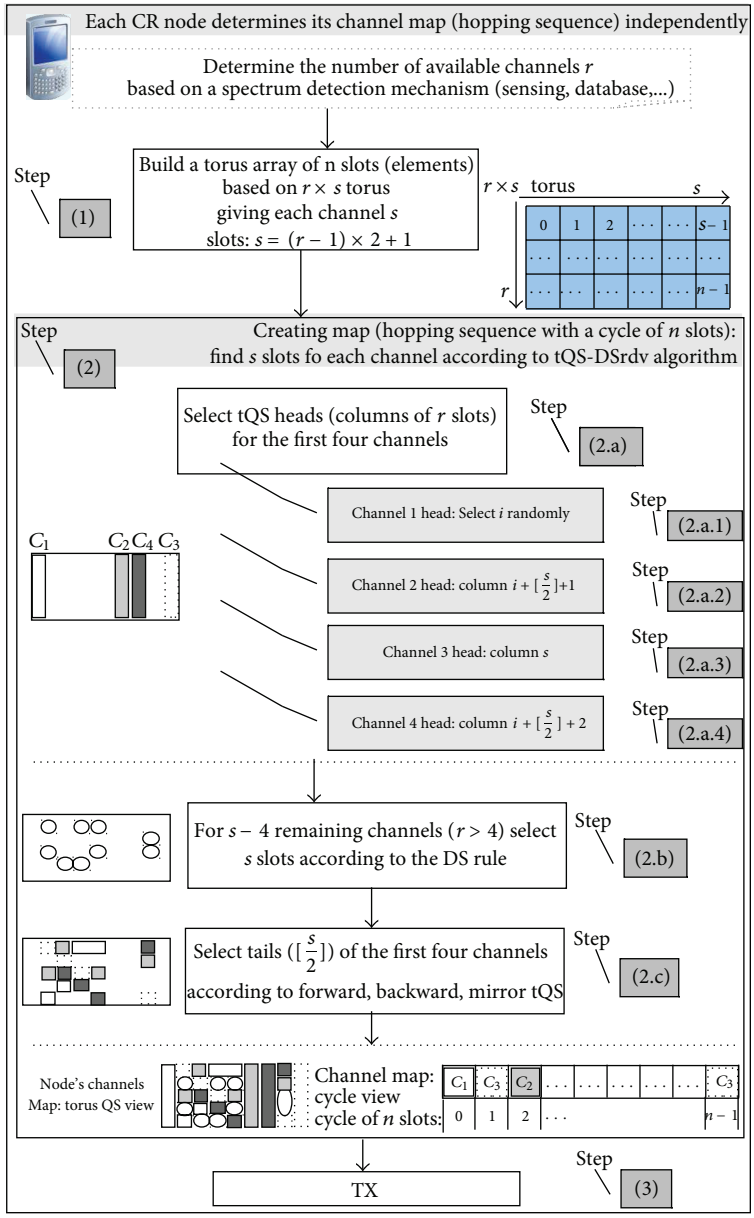


Figure 4: MtQS-DSrdv steps of each node.

construction. A node can select its head in s different ways because of s existing columns. In Figure 5 the head of Channel 1 is selected randomly to be C_1 (the first column). The remaining $s-4$ channels (step 2.b) search their map according to the DS rule (selecting a channel map by following the *DS rule* means that a relaxed DS set must be found of s slots; i.e., s remaining available elements are chosen so that at least one ordered pair (a_i, a_j) exists in (n, s) difference set, where $k = s$ from Definition 4) such that there are enough elements for tails of the first four channels.

For a node with eight available channels Figure 5 depicts two exemplary maps ($\text{Map}^{8\text{Chs}}(1)$ and $\text{Map}^{8\text{Chs}}(2)$) in order to show how to form another tQS-based map by just replacing the tail elements of the first four channels.

In addition, the reader should note that the selection of the channels is not strict; that is, we can replace tQS-based channels with DS based. Figure 6 illustrates a column selection of Channel 1 with different i (step 2.a.1 from Figure 4).

Thanks to the flexible use of tQS and DS concepts, the MtQS-DSrdv protocol allows automatically cognitive radio nodes to meet on each channel at least once in the symmetric channel view.

6. Verification

In the symmetric channel view we assess MtQS-DSrdv as regards the TTR performance. In the asymmetric channel

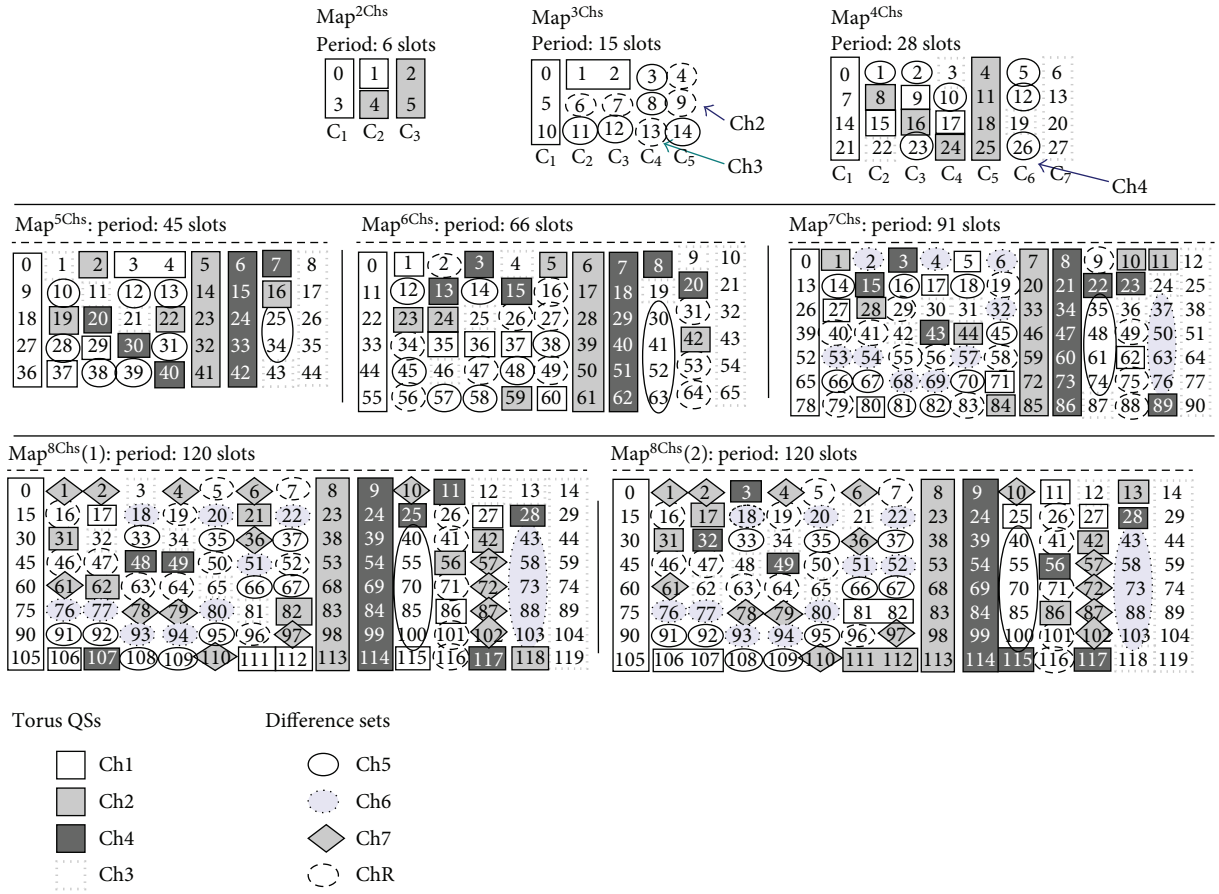


FIGURE 5: MtQS-DSrdv construction exemplary maps ($\text{Map}^{r\text{Chs}}$) for $r = 2 \cdots 8$ channels (Chs); Chr_i for $i = 1 \cdots r$ stands for a particular map of channel i. With three available channels, Channel 1 is selected according to the tQ forward manner, and the next two channels are formed from DSs ($\text{Map}^{3\text{Chs}}$), that is, DS_{Ch2}: {4, 6, 7, 9, 13} and DS_{Ch3}: {3, 8, 11, 12, 14}. These DSs are exemplary, since others could also be found. While having four available channels, Channel 4 is selected according to the DS rule. Starting with four available channels ($\text{Map}^{4\text{Chs}}$) the mirror tQ is utilized, changing twice the direction (k_i). With five available channels ($\text{Map}^{5\text{Chs}}$), Channel 2 is built using a backward tQ except of one mirror element.

view we evaluate the protocol in respect to the RDV guarantee and occurrence on each channel and TTR performance. The reader should note that we also compare the performance of the pair of nodes using maps with the same number of channels against the pair of nodes using maps with a different number of available channels. Hence, we analyze whether it is better to use a smaller map with only available channels or a larger map also with unavailable channels. The latter case refers to the case where unavailable channels are also visited, but only for spectrum sensing (this is a frequently used approach while dealing with ACHv in the related work [13, 26, 30]).

We define the Rotation Closure Property for a complete channel map (*map-RCP*) as follows.

Definition 5 (Map-Rotation Closure Property). For map $R1$ and $R2$ of period (cycle) $\Theta = \{0, \ldots, n-1\}$ and for all $i \in \Theta$, one defines for all slot offset, $\exists i : R1_i \cap R2_{(i+\text{slotOffset}) \bmod N} \neq \emptyset$, where slot offset $\in \Theta$.

The frequency map of a node must be checked with each possible cycle shift (slot offset $\neq 0$). With a slot offset 0 (cycle alignment case) nodes will always meet on all channels at least once.

Proof. A hopping pattern for each channel is constructed according to Definitions 1, 2, and 4; that is, each channel set is a torus quorum or cyclic difference set as shown in Figure 5. A torus quorum and cyclic DS satisfy the Rotation Closure Property from Definition 3. Therefore, the set of all channels, map $R1$, and $R2$ with the same period Θ, composed of elements from a $r \times s$ torus, so that $rs = n$, satisfies the map-RCP from Definition 5. We do not need to check all r channels because the map-RCP definition is automatically satisfied due to the fact that each single channel map satisfies the RCP or DS.

In all considered cases we show statistical results for the complete set (equal or different channels) of results obtained by two maps. We analyze the ACHv case with maximum

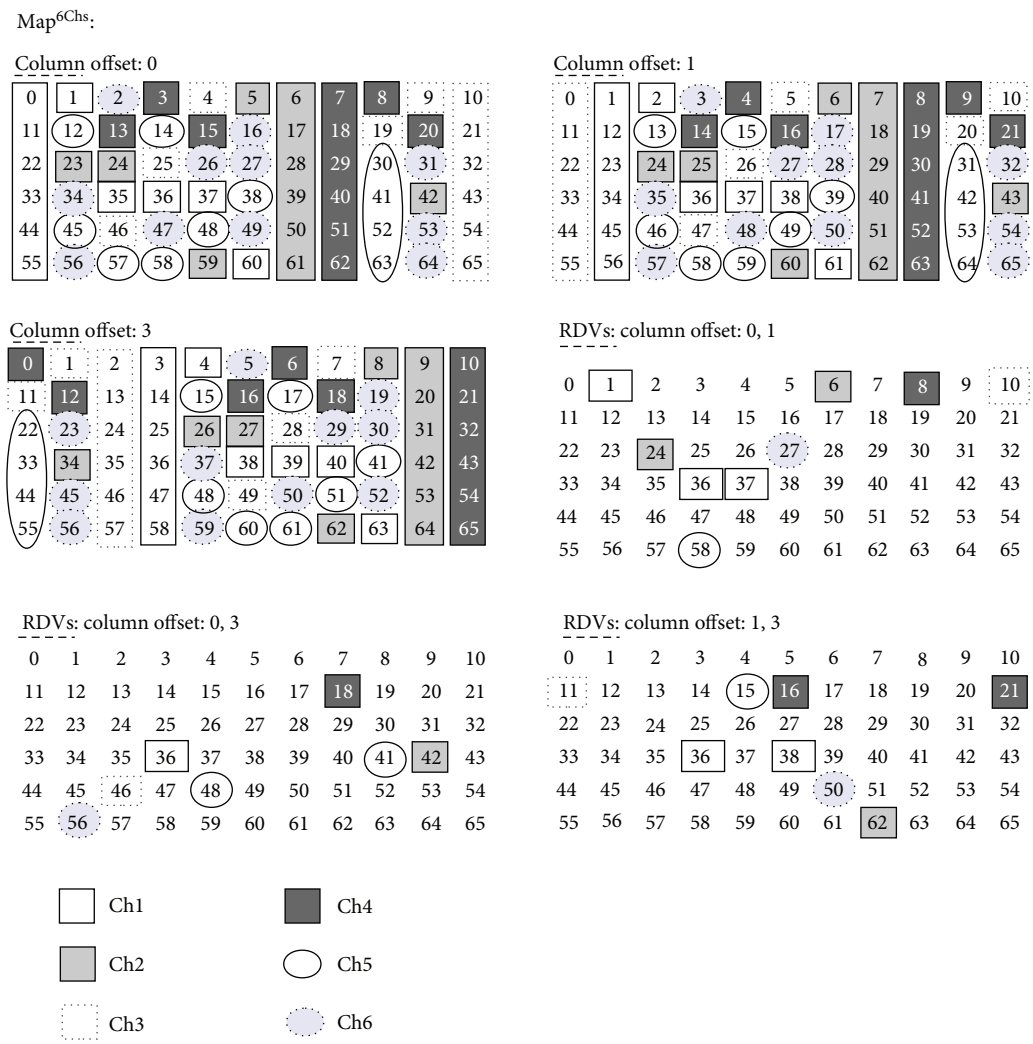


Figure 6: MtQS-DSrdv: different column offsets, CRs meet at least in r slots.

8 available channels, using channel maps from Figure 5. Moreover, while talking about the MtQS-DSrdv protocol, we refer to Figure 5, and Map$^{8\text{Chs}}$(1) for a map with eight channels unless mentioned otherwise. □

6.1. MTTR in Symmetric Channel View. In this subsection we evaluate MtQS-DSrdv in SCHv in terms of maximum TTR. Table 1 shows TTR results in terms of the minimum of all TTR maxima (Υ), the maximum TTR (MTTR), the mean (μ) of MTTRs, the mean (μ) of TTRs, and the mean of occurrences on a channel (μ^{Nch}) in a period.

In Table 1 we show the results for a map with 8 channels using both Map(1) ($8^{M(1)}$) and Map(2) ($8^{M(2)}$) from Figure 5. The reader should note that these are just two examples on how the tQS-based nodes can easily select their tail elements in a flexible manner.

In the case of SCHv it is clear that MTTR $< n$ ($n = |\Theta|$, where Θ is a period) for $r = 2-4$, and $\lesssim n/2$ for $r = 5-8$. One should note that we refer to MTTR per shift; that is, we take the maximum TTR in each shift, where μ^{MTTR} is an average of all MTTRs from all shifts while rotating the cycle. The average MTTR is considerably lower than MTTR. Note that the average TTR (μ^{TTR}) is significantly lower than MTTR and the average MTTR (μ^{MTTR}).

Table 1: MtQS-DSrdv in SCHv, so is *slot_offsets*; $(r, n) = \{(2, 6), (3, 15), (4, 28), (5, 45), (6, 66), (7, 91), (8, 120)\}$.

r	Υ	MTTR	μ^{MTTR}	μ^{TTR}	μ^{Nch}
2	1	4 (2 so)	3	1.5	1.2
3	4	9 (2 so)	5.7	2.5	1.4
4	3	18 (2 so)	10.3	3.5	1.6
5	4	21 (2 so)	12.6	4.5	1.6
6	5	35 (4 so)	20.6	5.5	1.7
7	6	47 (2 so)	23.6	6.5	1.7
$8^{M(1)}$	7	86 (2 so)	29	7.5	1.8
$8^{M(2)}$	7	57 (2 so)	30	7.5	1.8

In the worst case scenario while only one common channel is available and CRs can meet only once on the common channel, MTTR $= n$ (other less extreme cases

Table 2: MtQS-DSrdv in ACHv: RDV in a larger period of B; $_{\min}$RDV stands for the minimum number of RDVs (with the percentage).

A–B	$_{\min}$RDV	On all common CHs
2-3	1 (13%)	87%
2–4···8	2	100%
3-4···5	3	100%
3–6	2 (5%)	95%
3–7	3	100%
3–8	2 (7%)	93%
4-5	3 (16%)	84%
4–6	3 (5%)	95%
4–7···8	4	100%
5-6	3 (5%)	79%
5–7	3 (2%)	88%
5–8	2 (1%); 3 (5%)	77%
6-7	4 (3%)	73%
6–8	4 (1%)	78%
7-8	5 (10%); 6 (33%)	58%

are discussed in Section 6.3). However, as soon as a CR becomes aware of the fact that some channels are not available anymore, it can adjust its hopping pattern.

6.2. Minimum RDV Occurrence in Asymmetric Channel View. In this section we evaluate the MtQS-RDV protocol in the heterogeneous spectrum availability case, where some of the available channels of one CR are unavailable to another. While analyzing the ACHv case one should consider whether it is better to use a map with a larger period or to use a map with available channels only. For instance, in the former case, a CR with 3 currently available channels can select a map with 4 channels instead of 3, and while visiting an unavailable channel only spectrum sensing is performed. MtQS-DSrdv with a symmetric channel set, using a map for 4 channels while having 3 available channels, still guarantees RDVs on all 3 available common channels.

While using MtQS-DSrdv maps for a different number of channels than available, and the number of available channels is not large, there is also RDV guarantee on all available common channels most of the time (in a larger period). For instance, we compare $\text{Map}^{3\text{Chs}}$ (map with 3 available channels from Figure 5) of one node with $\text{Map}^{4\text{Chs}}$ of another node, but in a larger period of 28 slots as it happens when both nodes use $\text{Map}^{4\text{Chs}}$. In this case, nodes also meet on all 3 available channels.

The following definition considers the case where node A, which has a smaller period Θ, still meets with node B, which has a larger period Φ.

Definition 6 (nM-Maps-Rotation Closure Property). For map $R1$ with period $\Theta = \{0, \ldots, n-1\}$ and for all $i \in \Theta$, and map $R2$ with period $\Phi = \{0, \ldots, m-1\}$ and for all $j \in \Phi$, and where $n < m$, there is defined for all Φ $\exists i, \exists j : R1_i \cap R2_j \neq \emptyset$.

Table 2 shows the analysis of the combination of maps (with minimum $r = 2$ and maximum $r = 8$) while checking the number of RDVs in a larger period Φ. The results show that in many cases there is still RDV on all common channels of a node with fewer available channels. The probability to have an RDV on all shared channels is very high; for example, node with three available channels meets node with six available (3–6 case) at least on two channels in a period ($_{\min}\text{RDV} = 2$), and this can happen with only 5% probability, otherwise nodes meet on three common channels.

Let us next analyze an example where a node has four available channels in a set and another node has three available channels. In Figure 7 the first one (node $A1$) uses the map $\text{Map}^{4\text{Chs}}$ from Figure 5, and the second (node $A2$) uses $\text{Map}^{3\text{Chs}}$. While both nodes are aligned at slot 0, they meet on each channel in a smaller period on Channels 2 and 3 twice. We also show RDVs in a larger period; that is, apart from R1 rendezvous (in Φ we have $R1+R2$ RDVs) nodes meet in R2 slots (15, 18-19, and 22) on each channel. While shifting the alignment of a smaller period (Θ) map, we show RDVs only in a smaller period. The minimal number of channels on which a RDV occurs in a smaller period is one. Nodes usually meet on all three channels in a smaller period. Note that while analyzing RDVs in a larger period Φ of 28 slots, nodes have RDVs on *all* available channels of A2 in all shifts.

For the sake of clarity, we show in Figures 8 and 9 the RDV occurrence of $\text{Map}^{4\text{Chs}}$ in SCHv and a combination of maps with 3 and 4 channels. In both cases CR nodes meet on all three available channels. We can additionally notice that the combination of 3-4 channels maps is better than that using maps with 4 channels. Single RDV on a particular channel in a period is decreased significantly, since nodes have multiple RDVs in a period on all three channels. If we take a mean of RDV occurrence in a period for each channel, we obtain 1.8 with $\text{Map}^{4\text{Chs}}$ (SCHv) and 2.3 with the combination of maps (ACHv). In other words, it is better to use an asymmetric channel set in spectrum availability heterogeneity in this particular case.

In Figures 11 and 13 we show an example where nodes do not meet on each channel, namely, the combination of maps with 4 and 6 channels, and the combination of maps with 5 and 7 channels. In the former case, nodes do not have RDV on four channels ($_{\min}\text{RDV} = 3$) in 3 shifts (once on the first channel, and twice on fourth channel). However, as one can see in Figure 11, they meet very often, at least three times, on each channel in a period (a mean of RDV occurrence is circa 2.8). Contrary, while using a symmetric channel set map ($\text{Map}^{6\text{Chs}}$ in Figure 10) nodes meet on all channels, but most often once or twice in a period (a mean of RDV occurrence is circa 1.7). We also show an example while the pair of nodes uses $\text{Map}^{7\text{Chs}}$ (symmetric channel set) against the combination of maps with 5 and 7 channels (Figures 12 and 13). While using different maps (combination case), nodes have RDV guarantee on three channels ($_{\min}\text{RDV} = 3$), where RDV on three channels only happens in 2 shifts. RDV on four and five channels is the most frequent.

Although nodes do not meet on all channels, there is still a question whether this map should not be used, since nodes meet more often (mean of RDV occurrence is circa 2.6) than in the case of a symmetric channel set with 7 channels map (mean of RDV occurrence is circa 1.7).

Table 3: MtQS-DSrdv in ACHv: RDV in a larger period of B.

A–B	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
2-3	2	11 (3 so)	6.7	2.6	2.5
2–4	4	10 (5 so)	8.1	1.7	3.5
2–5	4	17 (7 so)	11	5	4.5
2–6	11	16	13	2	5.5
2–7	10	18	15.8	2.6	6.5
Map (1): 2–8	7	29	21.5	10.3	7.5
Map (2): 2–8	7	29	21	10	7.5
3-4	4	14 (4 so)	9.4	3.7	2.3
3–5	6	22 (3 so)	13	4.5	3
3–6	7	22 (4 so)	15	3	3.7
3–7	8	33 (6 so)	20.7	7	4.3
Map (1): 3–8	6	70 (8 so)	21.5	17	5
Map (2): 3–8	6	65 (8 so)	23	18	5
3^*–8^*	14	54 (8 so)	25	10	5
4-5	6	31 (2 so)	14	6	2.3
4–6	9	29 (2 so)	18	5.7	2.8
4–7	13	42 (4 so)	24.6	7	3.3
4–7^*	9	31 (3 so)	19	5	3.3
Map (1): 4–8	12	51 (4 so)	28.8	9	3.8
Map (2): 4–8	12	45 (4 so)	27.6	8	3.8
4–8^*	12	57 (4 so)	25	9	3.8
5-6	9	39 (1 so)	18	6.8	2.2
5-6^*	6	32 (3 so)	17	6	2.2
5–7	8	45 (2 so)	22.7	7	2.6
5–7^*	9	47 (2 so)	21	9	2.6
Map (1): 5–8	12	65 (3 so)	26.7	9.4	3
Map (2): 5–8	10	70 (3 so)	27	11	3
5–8^*	10	56 (3 so)	24	10	3
6-7	10	66 (1 so)	22.9	8.8	1.7
6^*-7^*	10	43 (2 so)	19	7	2.2
Map (1): 6–8	11	51 (2 so)	26	7.6	2.5
Map (2): 6–8	10	49 (2 so)	25.8	8	2.5
6^*–8^*	8	46 (2 so)	25	8	2.5
Map (1): 7-8	13	55 (2 so)	27.7	9.5	2.2
Map (2): 7-8	9	64 (1 so)	25	9	2.2
7^*-8^*	12	58 (1 so)	26	10	2.2

6.3. MTTR in Asymmetric Channel View. In Table 3 MTTR statistics are presented for the asymmetric channel view. One should note that we refer to MTTR per shift (slot offset, "so" in table); that is, we take the maximum TTR in each shift, where μ^{MTTR} is an average of all MTTRs from all shifts while rotating the cycle.

In ACHv $A - B$ denotes combinations of maps of node A with node B, where the number of channels of node A is smaller than the number of channels of node B, as defined before. In this table we also show the results for a map with 8 channels using both $\text{Map}^{8\text{Chs}}(1)$ and $\text{Map}^{8\text{Chs}}(2)$ from Figure 5. In addition, in some of the combinations we have shown two cases, with or without (*). The case without (*) stands for the maps as proposed in Figure 5; that is, the first channels are assigned to tQS maps, and the next to DS maps. In the case with (*), in the map where there are more DSs than one (with $r = 3$ and $r = 6 \ldots 8$), the first channels are assigned to DSs (following the order from Figure 5, thus, map of Ch1 becomes Ch5, then Ch2 $\leftarrow$ Ch6, Ch3 $\leftarrow$ Ch7, and Ch8 $\leftarrow$ Ch4), where the remaining channels are assigned to tQS-based maps, also following the order from Figure 5 (e.g., map Ch5 $\leftarrow$ Ch1). Note that the described results below are without (*), unless mentioned otherwise in the text.

For ACHv the situation is naturally different than that for SCHv (Table 1) due to different period sizes of the compared channel sets. The results below are shown from the perspective of larger period (Φ) in order to compare it with the SCHv case. Depending on the combination of the

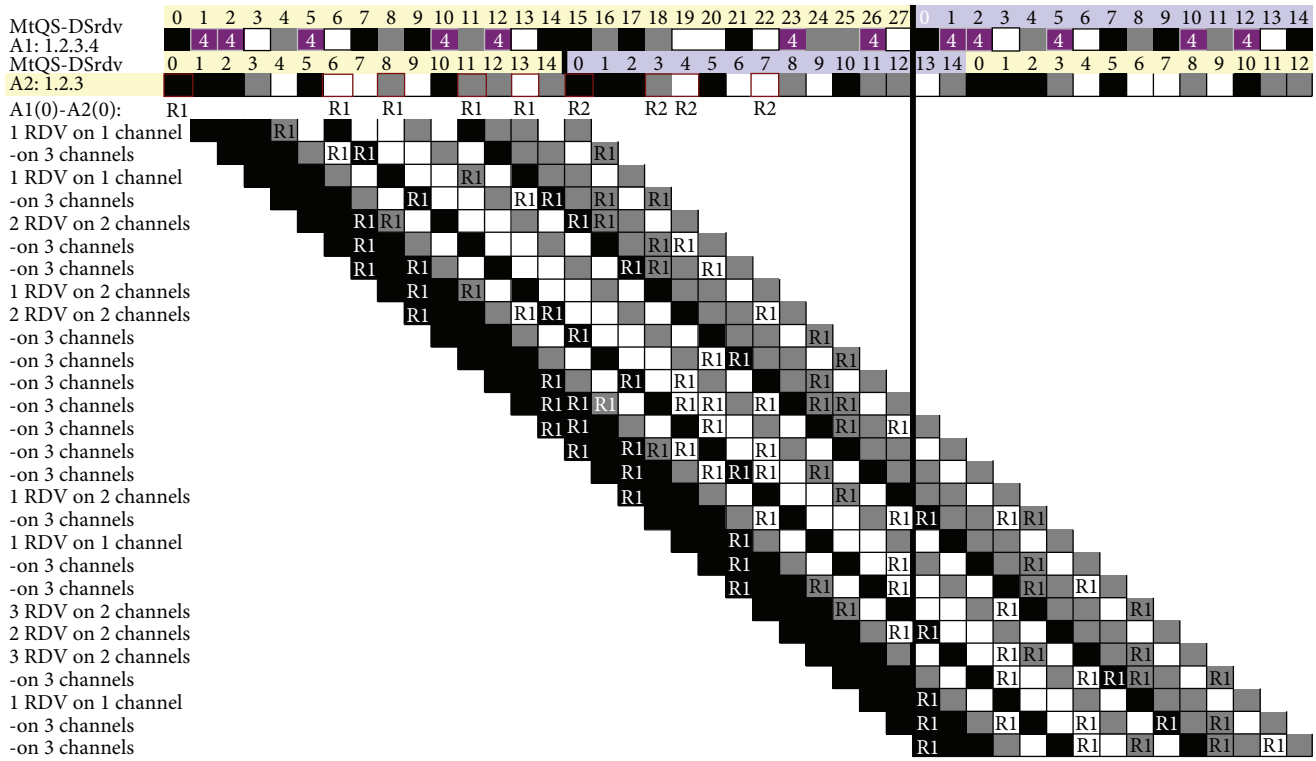


FIGURE 7: Asymmetric channel view: combination of maps with 3 and 4 channels, while rotating the cycle; RDV consideration in Θ; *black* square stands for channel 1, *gray* square for channel 2, and *white* square for channel 3; R stands for Rendezvous.

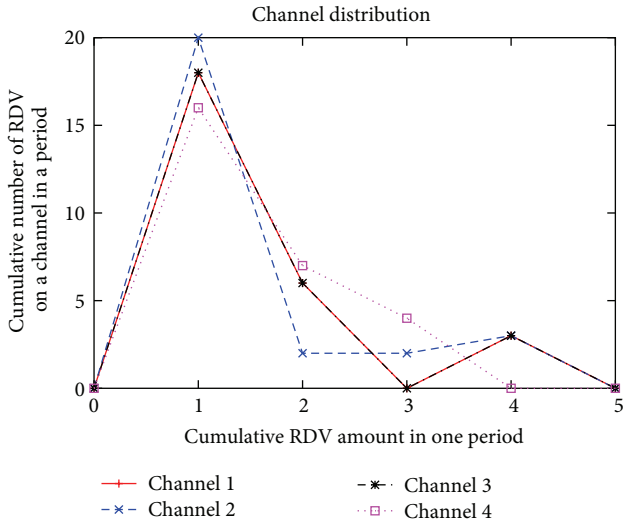


FIGURE 8: Symmetric channel set: Map^{4Chs} (28 slots).

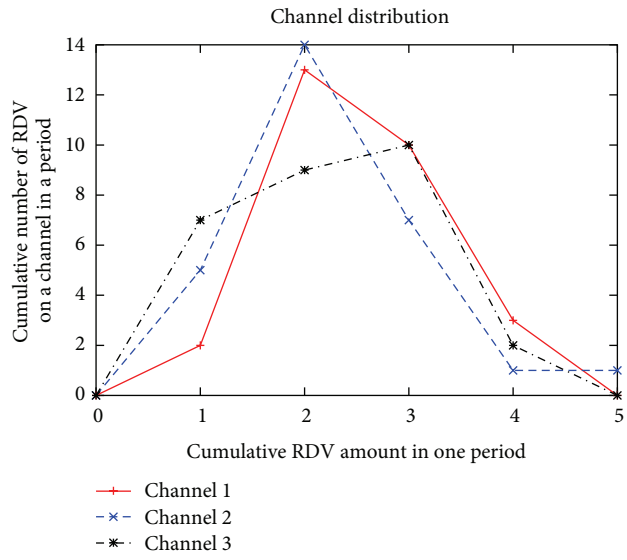


FIGURE 9: Asymmetric channel set: combination of maps with 3 and 4 channels.

channels of nodes A and B, the maximum TTR varies usually between $m/2$ and $(m/3)$ ($m = |\Phi|$), where the average MTTR is lower than 30 slots in all cases, varying usually between $m/3$ and $m/4$. We also show some of the combinations where DS channel maps are reordered with tQS channel maps, the case with (*). One can see that assigning first DS maps and then tQS maps to channels usually decreases MTTR and/or MTTR mean. We observe that the selection of different tail elements (as exemplary shown with maps Map^{8Chs}) can improve MTTR or μ^{MTTR}.

Cognitive radio nodes that use a different map increase significantly RDV occurrences (μ^{Nch}) on a channel in a period. The reader should note that in comparison with the performance in SCHv, RDV occurrence only once in a period

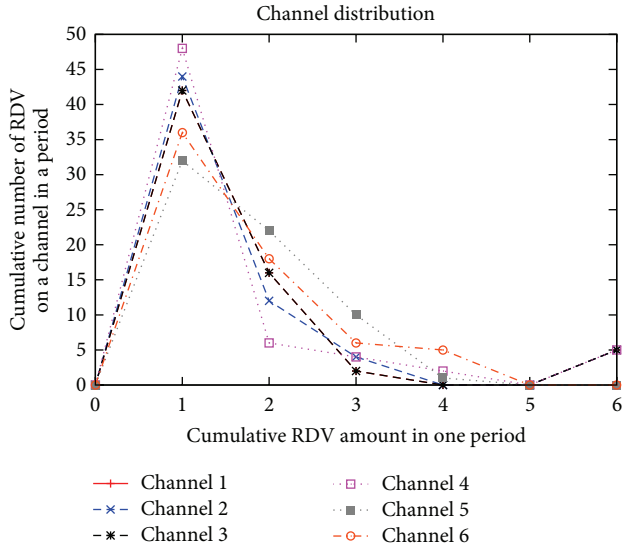


FIGURE 10: Common channel set: Map^{6Chs} (66 slots).

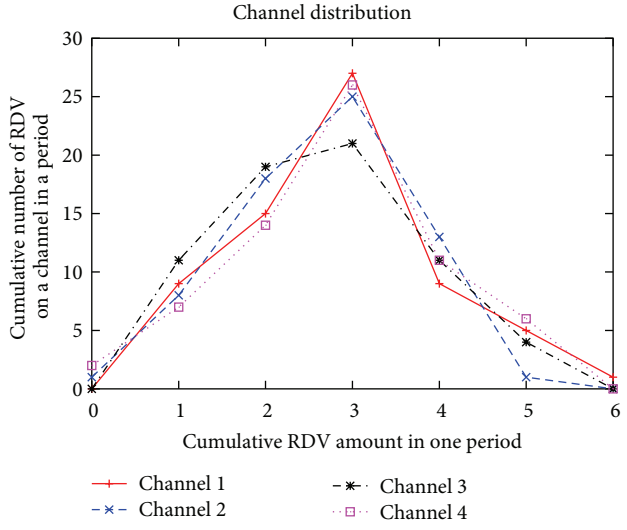


FIGURE 11: Asymmetric channel set: combination of maps with 4 and 6 channels.

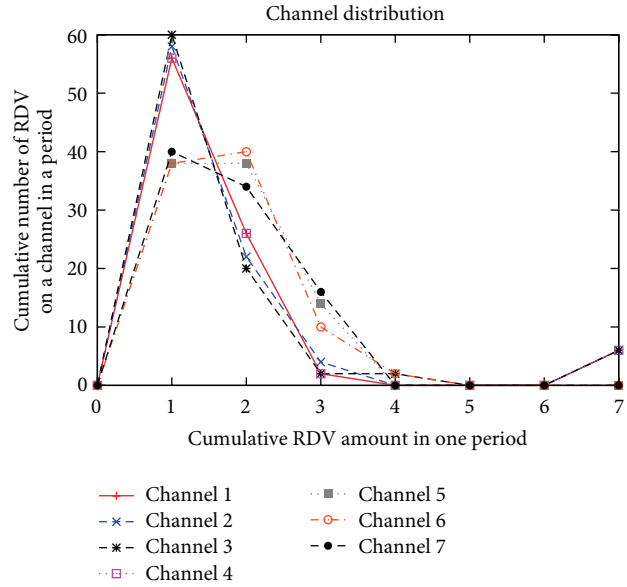


FIGURE 12: Common channel set: Map^{7Chs} (91 slots).

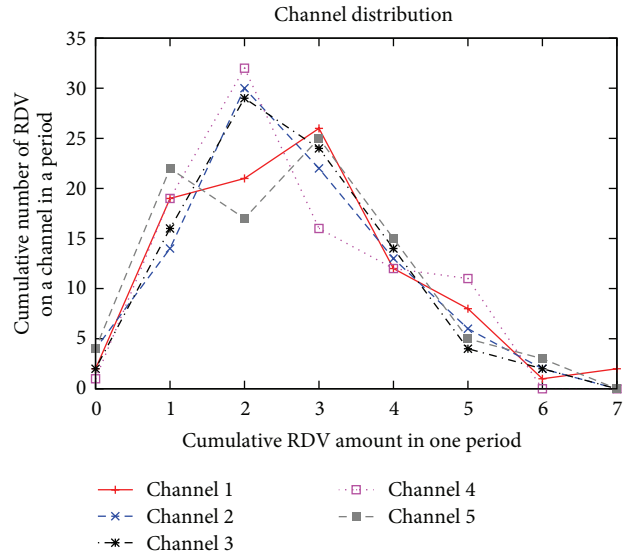


FIGURE 13: Asymmetric channel set: combination of maps with 5 and 7 channels.

is insignificant, since multiple RDVs are the dominant case. However, in order to answer the question which map one should use, the one imposing a symmetric channel view, or the one using an asymmetric channel set, the comparison of Tables 1 and 3 will not provide conclusive answer. Therefore, in Table 4 we show the MTTR statistics in respect to a smaller number of available common channels κ, although using maps with a larger number of channels.

Note that in this implicit way the related work papers handle an asymmetric view case [13, 26, 30]. In this case it is also easy to verify an upper bound of MTTR of the MtQS-DSrdv design (due to the RCP guarantee CRs always meet on every channel in a sequence period), which will be MTTR $\leq (m - \kappa)$ in the worst case scenario (the worst possible chosen maps). For example, with $r = 8$ and $\kappa = 2(\mathbf{8} \Rightarrow 2)$ such upper bound is met in two cycle shifts (2 so in the table); since MTTR = 118 which is exactly $(m - \kappa)$, the same situation is in cases, $r = 3$ and $\kappa = 2$, $r = 6$ and $\kappa = 2$, and $r = 7$ and $\kappa = 2$. In other cases MTTR is lower or significantly lower.

Combining the results from Tables 3 and 4 it is clear that in terms of (M)TTR it is more advantageous to use a map with a number of locally available channels instead of using a larger map. MTTR of asymmetric channel set maps is always lower except for a few cases and even in those exceptional cases μ^{MTTR} of asymmetric channel set maps is better than that of symmetric channel set maps. Nevertheless, an investigation should be done combining the selection of a map including the duration of nonavailability of channels. It might be better

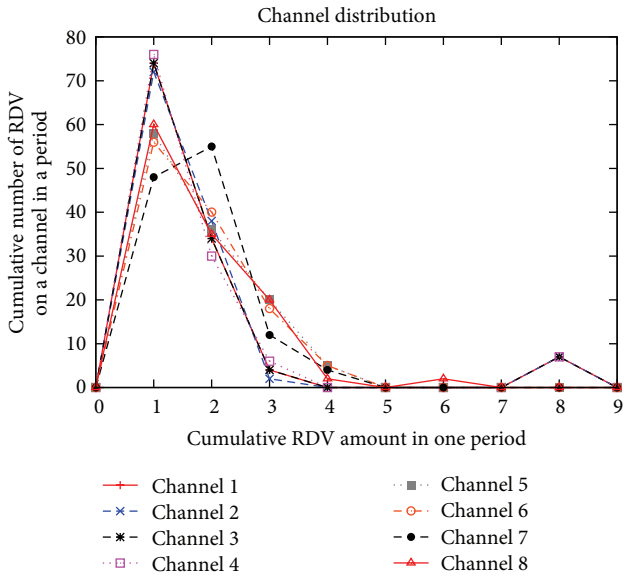


Figure 14: MtQS-DSrdv: 8 channels (120 slots).

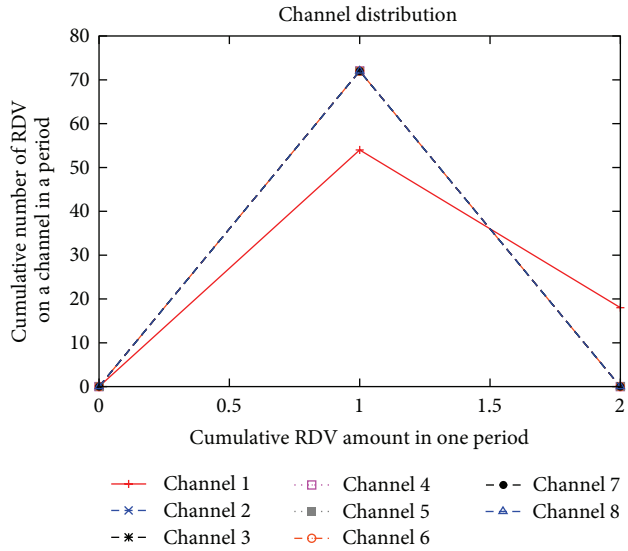


Figure 15: DSMMAC [13]: 8 channels (73 slots).

to use a larger map keeping in mind a probability of the channel release (predicting channel availability based on past observations) of the channel occupied by a PU recently. Hence, an RDV protocol should closely cooperate with a spectrum decision mechanism, which will be the topic of interest in our future work.

6.4. Comparison with the Related Work. In the subsection we compare the MtQS-DSrdv protocol with the related work where the designed approaches guarantee RDV either on all channels or in a very small TTR time. Hence, we consider the DSMMAC (*Difference-Set*-based asynchronous Multichannel MAC) [13], A-MOCH (Asynchronous Maximum Overlapping CH protocol) [30], ASYNC-ETCH (Asynchronous Efficient Channel Hopping) [31], and Balanced Incomplete Block Design- (BIBD-) based [29] algorithms in the synchronous channel view and consider also asynchronous channel view if applicable. In all considered cases we show results including the rotation of a whole cycle; that is, all possible cases without cycle alignment are checked.

6.4.1. Comparison with DSMMAC. In Figure 14 we depict results of MtQS-DSrdv with 8 channels according to the map presented in Figure 5. Figure 15 illustrates DSMMAC results with 8 channels. We remind the reader that the map from [13] is constructed entirely from Difference Sets. MTTR of MtQS-DSrdv equals 86 slots (using Map(1), but 57 slots with Map(2)), which happens in two shifts only. The average MTTR (μ^{MTTR}) is 28.8 with a standard deviation of 12. However, one should note that nodes meet often multiple times ($\mu^{\text{Nch}} = 1.8$) in a period as it can be seen in the figure. In this particular example the MTTR of DSMMAC is better than that of MtQS-DSrdv, since it equals 44 slots, with a mean of 23 slots and a standard deviation of 8.1. However, as can be clearly seen in Figure 15, on DSMMAC channels $2 \cdots 8$, nodes always meet only *once* in a period and *no more*. In DSMMAC, Channel 1 gets an extra slot, since all DSs have been chosen from a period of 73 slots, having 9 elements in a set. The remaining slot (slot 0) has been assigned to Channel 1, and therefore on Channel 1 nodes also have RDV twice in a period (Figure 15).

The fact that nodes meet only once in a period on $2 \cdots 8$ channels diminishes the chance of the protocol to be adaptable in ACHv, which was dismissed in this work. In [13] it is assumed that every node should use the same hopping sequence. Therefore, we also check the MTTR statistics while both CRs apply 8 channel maps, but they can meet only on 2 common channels because only these two are available for one of the CRs. The MTTR of DSMMAC will increase until 71 slots (which happens twice), whereas μ^{MTTR} is increased to 50.4 with $\sigma^{\text{MTTR}} = 10.4$. Those MTTR results are clearly much worse than using the 2–8 combination of maps of MtQS-DSrdv (Table 3).

We also analyze the possibility of the combination of maps with 2 and 8 channels for this approach. Figures 16 and 17 show the results for both protocols.

With both protocols nodes meet on each channel. The MTTR of DSMMAC is slightly lower (25 slots) than that of MtQS-DSrdv (29 slots, see Table 3). μ^{MTTR} is similar (20.5 of DSMMAC), but μ^{Nch} of DSMMAC is lower than that of MtQS-DSrdv (for both channels 7.5), since the average RDV occurrence on Channel 1 equals to 4.4 and on Channel 2 is 5.1, which can also be observed from the figures.

The DSMMAC map with eight available channels has seven neighboring pairs of slots, one of each channel ($2 \cdots 8$), except of Channel 1 having once three neighboring slots. Whereas, the MtQS-DSrdv map has more neighboring slots and can have even more thanks to the flexible design of the protocol. This characteristic is paramount with asynchronous nodes (without slot alignment), since nonoverlapping slots influence badly the chance for actual RDV

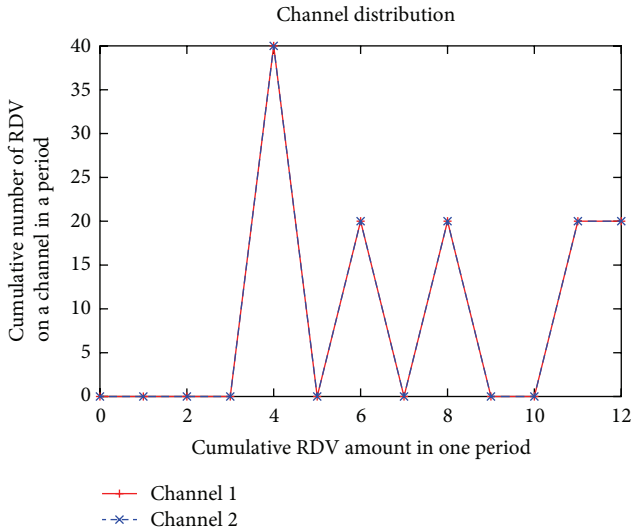


Figure 16: MtQS-DSrdv in ACHv: the combination of maps with 2 and 8 channels (120 slots).

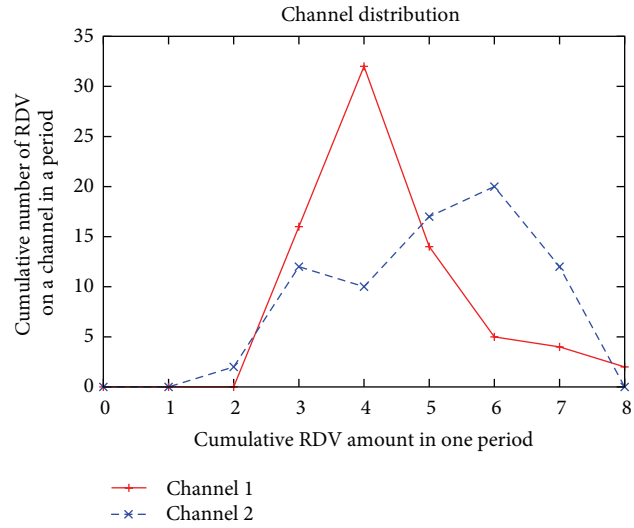


Figure 17: DSMMAC [13]: the combination of maps with 2 and 8 channels (73 slots).

on a channel. Hence, the MtQS-DSrdv map predominates DSMMAC in this context. However, further comparison of both approaches is difficult due to the unavailability of maps with other number of channels.

6.4.2. Comparison with A-MOCH. In Table 5 we show the results of the A-MOCH protocol [30] for $r = 2 \cdots 8$ channels. We remind that in the A-MOCH algorithm a sender generates its hopping sequence based on the LS array and the receiver map is based on the IRS array. Since according to the algorithm nodes may randomly select a permutation of $\{0, 1, \ldots, N-1\}$, where N denotes the number of channels, we created all the maps in the same way, that is, rotating each row in a forward wrap-around manner to receive easily an LS square. For instance, with 5 channels the sender channel hopping sequence in the example below is {0, 1, 2, 3, 4, 4, 0, 1, 2, 3, 3, 4, 0, 1, 2, 2, 3, 4, 0, 1, 1, 2, 3, 4, 0}. The receiver with 5 channels adopts the following sequence: {0, 1, 2, 3, 4, 0,1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4}. This map construction gave us the worst possible map cases, since the mean MTTR is always equal to the MTTR and is constant as shown in the table; that is, there are no differences between MTTR, Υ (minimum of MTTR maxima), and μ^{MTTR}. Thus, the meeting points will always have the same occurrence (MTTR $= N^2 - N + 1$ as reported in [30]). In addition, nodes will always meet *once* on each channel in each period (μ^{Nch}). In comparison with MtQS-DSrdv, the MTTR of A-MOCH is slightly better, we say slightly, since MTTR of MtQS-DSrdv happens to be maximum in 4 cases (usually 2 cases), whereas for A-MOCH for every shift. Of course as a consequence, the mean (μ^{MTTR}) of A-MOCH is significantly worse than of MtQS-DSrdv.

As mentioned above, the table shows the worst cases of MTTR. If we select other sequences we can improve the MTTR, for example, with 5 available channels and the sender adopting the following sequence: {0, 1, 2, 3, 4, 1, 2, 0, 4, 3, 2, 3, 4, 0, 1, 3, 4, 1, 2, 0, 4, 0, 3, 1, 2}, and the receiver with the sequence:{4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 0, 1, 2, 3}, we can improve μ^{MTTR} to 15 with $\sigma^{\text{MTTR}} = 5.4$ and $\Upsilon = 8$. The MTTR stays 21, but it occurs 10 times.

The reader should note that μ^{Nch} can never be changed (independently of chosen sequence) and remains always *one* RDV on each channel in a period, which diminishes the RDV chances of asynchronous nodes (partial overlap of a single slot might not be sufficient to have a successful RDV).

Moreover, the protocol does not work explicitly with the heterogeneous spectrum availability, since if we take the sender map with, for example, 5 channels and the receiver with, for example, 8 channels, then there is no RDV guarantee, even in a larger period where there are *eight* shifts with no matches. Therefore, the only possible option is to use the map with 8 channels, that is, also hopping on unavailable channels. For instance, if the $\kappa = 5$, but $r = 8$, then the MTTR of A-MOCH equals 60 (occurring 32 times), with $\mu^{\text{MTTR}} = 58.5$, which is automatically worse than MtQS-DSrdv with the combination of maps with 5 and 8 channels (5–8 in Table 3).

If we take $\kappa = 2$ and $r = 8$ for A-MOCH, then we obtain MTTR = 63 (56 times) with $\mu^{\text{MTTR}} = 62.3$, which is definitely worse than the results of MtQS-DSrdv with the combination of maps with 2 and 8 channels (2–8 in Table 3).

Taking a better (aforementioned) map with 5 channels, and checking the least extreme case, namely, with $\kappa = 4$, then we get MTTR = 22 with a mean of 15.4. The MTTR is somewhat better than that of MtQS-DSrdv, but the mean is similar. However, with $\kappa = 3$, MTTR is 23 (5 times) and mean 19, which is already worse than the asymmetric channel set with 3 and 5 channels' combination of maps of MtQS-DSrdv (3–5 in Table 3).

In [30], the metric Maximum Conditional TTR (MCTTR) has been defined which refers to the maximum TTR between two hopping nodes when PU traffic is present and at least one channel is available for the two nodes.

Table 4: MtQS-DSrdv: maximum time-to-RDV performance of κ.

$r \Rightarrow \kappa$	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
3$\Rightarrow$ 2	4	13 (2 so)	7.3	3.2	1.4
4 $\Rightarrow$ 2	3	26 (2 so)	15.6	6.6	1.6
4 $\Rightarrow$ 3	3	22 (2 so)	12.4	5.3	1.6
5 $\Rightarrow$ 2	4	41 (2 so)	25.2	9.5	1.6
5 $\Rightarrow$ 3	4	33 (2 so)	19	6.6	1.6
5 $\Rightarrow$ 4	4	27 (4 so)	16.4	6.3	1.6
6 $\Rightarrow$ 2	5	64 (2 so)	41	15.9	1.7
6 $\Rightarrow$ 3	5	55 (8 so)	34	13.8	1.7
6 $\Rightarrow$ 4	5	51 (2 so)	30	11.8	1.7
6 $\Rightarrow$ 5	5	42 (4 so)	24.2	9.3	1.7
7 $\Rightarrow$ 2	6	89 (2 so)	57.7	20.5	1.7
7 $\Rightarrow$ 3	6	82 (2 so)	44.4	16.3	1.7
7 $\Rightarrow$ 4	6	63 (2 so)	38.3	13.9	1.7
7 $\Rightarrow$ 5	6	57 (2 so)	31	11.7	1.7
7 $\Rightarrow$ 6	6	53 (2 so)	26.8	9.9	1.7
8 $\Rightarrow$ 2	7	118 (2 so)	73	25.6	1.8
8 $\Rightarrow$ 3	7	101 (2 so)	57	21.4	1.8
8 $\Rightarrow$ 4	7	101 (2 so)	47.8	19.8	1.8
8 $\Rightarrow$ 5	7	93 (2 so)	39.7	16.4	1.8
8 $\Rightarrow$ 6	7	93 (2 so)	35.6	15.3	1.8
8 $\Rightarrow$ 7	7	86 (2 so)	33	14.9	1.8

Table 5: A-MOCH [30] in SCHv $(r, n) = \{(2,4), (3,9), (4,16), (5,25), (6,36), (7,49), (8,64)\}$.

#	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
2	3	3 (4 times)	3	0	1
3	7	7 (9 times)	7	0	1
4	13	13 (16 times)	13	0	1
5	21	21 (25 times)	21	0	1
6	31	31 (36 times)	31	0	1
7	43	43 (43 times)	42	0	1
8	57	57 (57 times)	57	0	1

MCTTR of A-MOCH is at least N^2 (N denotes the number of channels), thus, with 5 channels MCTTR = 25, and with 8 channels MCTTR = 64. Coming back to our protocol, if we know that only one channel is incumbent-free for a long period of time (such information can be obtained for instance from Radio Environmental Maps), and we cannot visit other channels, then the MCTTR will equal s, for example, for 5 channels $s = 9$, for 8 channels $s = 15$.

6.4.3. Comparison with ASYNC-ETCH. In Figures 18 and 19 we show the cumulative channel distribution of A-ETCH [31] in a hopping period for all shifts with 5 available channels. We remind that in A-ETCH a hopping sequence S_i is composed of N frames (N denotes the number of available channels), where each frame is composed of a pilot slot and two subsequences sub S_i. The pilot slots of S_i, collected together, are the channels appearing in sub S_i in the same order, thus $|S_i| = (2N + 1) * N$; for example, with 5 channels a period is composed of 55 slots in total since $|S_i| = (2 * 5 + 1) * 5 = 55$. Figure 18 depicts the case when nodes select the same sequence, where we show the following sequence (S_0): {0, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 1, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 2, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4, 3, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4 , 4, 0, 1, 2, 3, 4, 0, 1, 2, 3, 4}. Figure 19 depicts the case when one CR selects (S_0) and another (S_1): {0, 0, 2, 4, 1, 3, 0, 2, 4, 1, 3, 2, 0, 2, 4, 1, 3, 0, 2, 4, 1, 3, 4, 0, 2, 4, 1, 3, 0, 2, 4, 1, 3, 1, 0, 2, 4, 1, 3, 0, 2, 4, 1, 3, 3, 0, 2, 4, 1, 3, 0, 2, 4, 1, 3}.

The average TTR (μ^{TTR}) is reported to be $(2N^2 + N)/(N - 1) \approx 2N$. Hence, with $N = 5$ we have $\mu^{\text{TTR}} \approx 11$, which is indeed the case but with two different maps (see Figure 19). If the pair of CRs selects the same hopping sequence, RDV is guaranteed in one slot in a hopping period. Using different sequences RDV is guaranteed in N slots.

The reader should note that this guarantee does not happen on all channels, which is clearly noticeable in the figures. With *different* maps, there are shifts where RDV happens on one or two channels. As a consequence, it might happen that there is indeed RDV guarantee on even 11 or 12 slots of 55 slots, but only on 1, 2, or 3 channels, which is highly undesirable, especially in the case when these channels are of bad quality. Moreover, RDV on particular channels is more random, since it depends on the shift and sequence. To be more precise, there are 40 shifts where nodes meet on *three* channels, but they will *never* meet on *four* or *five* channels in one hopping period. What Figure 19 also shows is that the RDV frequency on a channel is alternating from very low (once or twice) to moderate (5 or 6 times) or to very high (10 times) with this scheme.

In the case that the nodes select the *same* hopping sequence (Figure 18) MTTR is 54 slots with the mean value of

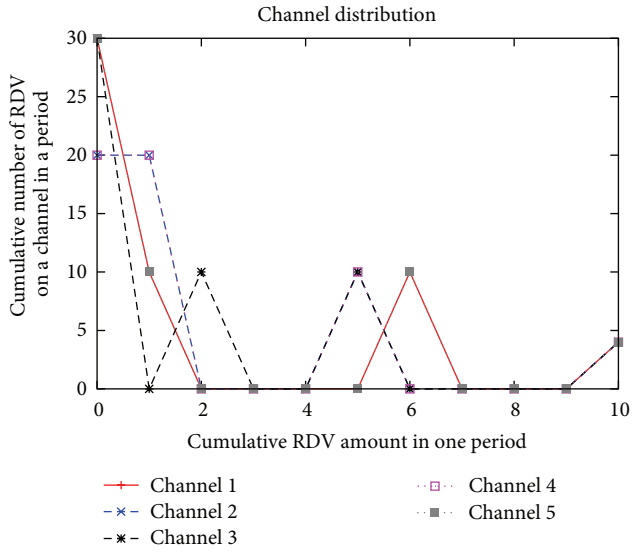


Figure 18: A-ETCH with the same sequences (5 channels).

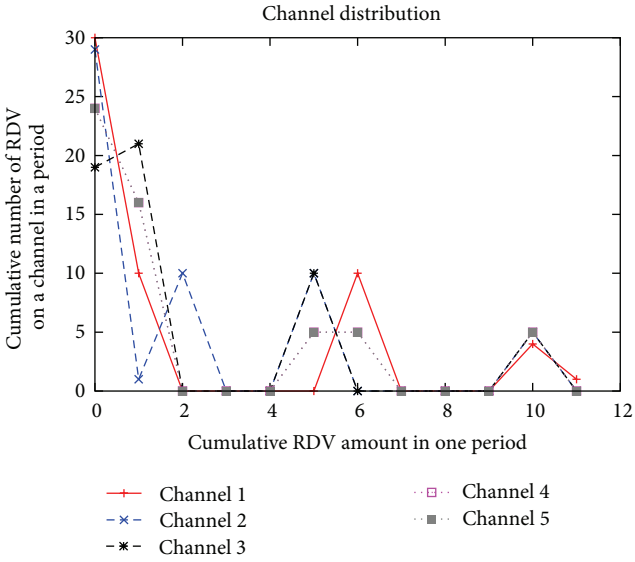


Figure 19: A-ETCH with the different sequences (5 channels).

32 slots. An interesting observation is that in this case there are 14 shifts in which nodes meet on all *five* channels in one hopping period, but on the remaining shifts (40) they meet only on 1 or 2 channels.

We also compare A-ETCH in ACHv, using the order of maps as in S_0 and S_1 (i.e., $i = 0, 1$ in Algorithm 2 from [31]) in Table 6. The rows without (*) stand for the case when nodes select the same maps S_0 (but of course for a different number of channels N). The rows with (*) show results with the same S_1 map, and finally information with (**) is about selection of different sequences (S_1 and S_0).

One observes that having maps with a different number of channels, allow nodes to meet always on all channels in a period. However, the choice of the maps is extremely influential on MTTR statistics, since it can be much worse

Table 6: A-ETCH in ACHv with S_0 sequence combinations for $N = 3, 5, 7$; $(r, n) = \{(3, 21), (5, 55), (7, 105)\}$.

#	Υ	MTTR	μ^{MTTR}	σ^{MTTR}	μ^{Nch}
3–5	8	36 (9 times)	21	9	3.6
(*) 3–5	8	18 (20 times)	14	4	3.6
(**) 3–5	6	12 (19 times)	10	2	3.6
3–7	15	51 (25 times)	38	10	5
(*) 3–7	15	26 (60 times)	24	3	5
(**) 3–7	9	24 (25 times)	17.5	4	5
5–7	19	79 (14 times)	44	20.4	3
(*) 5–7	19	40 (8 times)	26.5	8	3
(**) 5–7	10	17 (48 times)	14	3	3

than that of MtQS-DSrdv (the case without star, i.e., S_0 maps combinations), similar to that of MtQS-DSrdv (the case with (*), i.e., S_1 maps combinations), and better than that of MtQS-DSrdv (combination of S_1 and S_0 maps). Hence, the algorithm from [31] has room for improvement, especially in the case while nodes have the same channel view (SCHv), since TTR $\approx 2N$ is definitely not sufficient if nodes meet on one or two channels while having much more in a set. In ACHv there is a need for a kind of avoidance of the selection of the same sequence in order to improve TTR statistics. Finally, the algorithm assumes that N must be prime, but it can be also generalized for a nonprime N.

6.4.4. Comparison with Balanced Incomplete Block Design-Based RDV Approach. In order to guarantee RDV the Balanced Incomplete Block Design (BIBD) (a BIBD [35] is an arrangement of v distinct objects into b blocks, so that each block contains exactly k distinct objects, each object occurs in exactly r different blocks, and every pair of distinct object a_i, a_j occurs together in λ blocks. A BIBD has five parameters $\{v, b, r, k, \lambda\}$ so that $bk = vr$ and $r(k-1) = \lambda(v-1)$. While $v = b$, and then $k = r$ then such BIBD is called *symmetric* using three parameters $\{v, k, \lambda\}$) feature has been used in [29]. With two available channels a BIBD of $\{7, 3, 1\}$ is used in [29]. In order to have an RDV guarantee for multiple channels, all channels C are divided into two groups, $G1 = \{1, 2, \ldots, C/2\}$ and $G2 = \{C/2 + 1, (C/2) + 1, \ldots, C\}$. Each group is assigned to a different BIBD state; for example, BIBD of $\{7, 3, 1\}$ has $\{1, 1, 0, 1, 0, 0, 0\}$. Hence, we obtain $\{G1, G1, G2, G1, G2, G2, G2\}$. Each group is divided into two subgroups, for example, $G1 = \{G1.1, G1.2\}$ and $G1 = \{G2.1, G2.2\}$. Afterwards, each subgroup is assigned again to a different BIBD state; for example, using the same BIBD sequence, $G1$ is assigned into $\{G1.1, G1.1, G1.2, G1.1, G1.2, G1.2, G1.2\}$ and $G2$ into $\{G2.1, G2.1, G2.2, G2.1, G2.2, G2.2, G2.2\}$. The steps (iterations) are repeated until each channel has a subgroup. This algorithm is indeed very simple, and it is very easy to form an RDV sequence using, for example, the same BIBD sequence. However, one must note than there are also shortcomings of this scheme; first, it is not a problem to assign 2^s channels where $s = \{1, 2, 4, 5, \ldots\}$, since this number of channels is easy to divide multiple times in groups according the aforementioned algorithm. However, if

$n = \{3, 5, 6, 7, \ldots\}$, some of the channels must be assigned to both leaf groups (the final last subgroups) in order to form a symmetric BIBD satisfying intersections of all channels in all blocks (RCP using QS context). Moreover, the *cycle* of such sequences increases very fast. In [29] it is stated that MTTR = v^i, where v is the number of building blocks (slot offsets or shifts), where i is the number of building iterations of this sequence and $i = \log_2 C$ (C number of channels). Therefore, with *two* channels we have MTTR = 7 for aforementioned example of BIBD ($\{7, 3, 1\}$), with four channels MTTR = 7^2 = 49, with *eight* channels MTTR = 7^3 = 343, and with *sixteen* channels MTTR = 7^4 = 2401. MTTR is also equivalent to the size of the whole sequence, that is, the size of a *cycle*; that is, with *sixteen* channels the sequence has 2401 slots!

Both protocols, MtQS-DSrdv and BIBD based, guarantee RDV on all channels in the period defined by the corresponding approach. If we compare in terms of MTTR, for example, the case with available 4 channels, the cycle of MtQS-DSrdv is much lower (28 instead 49 slots) and MTTR is also much lower, 18 slots with a mean of 10 slots, instead of 49 slots; the case with 8 available channels, the cycle of MtQS-DSrdv is significantly lower (120 instead of 343 slots) and MTTR significantly lower, 86 slots with a mean of 29 slots, instead of 343 slots.

Analyzing BIBD approach in the case with heterogeneous spectrum availability, for example, with one node with 4 available channels and the other with 8 available channels, we observe than nodes still meet on four channels of the first node, with MTTR equal to 172 slots, whereas with MtQS-DSrdv we have MTTR of 51 slots with a mean of 29 slots, which is significantly better than the BIBD approach.

7. Conclusion and Future Work

In this work we evaluated a torus-QS- and DS-based rendezvous protocol while having symmetric and asymmetric channel views. We showed that the heterogeneous spectrum availability does not decrease the performance of our RDV protocol, since nodes either can still meet on all available common channels in each period, or they meet multiple times on the visited common channels. MTTR is bounded and usually being smaller than half of a larger period. We have also shown that our algorithm is more efficient and more stable in comparison with the related approaches. We pointed out that a small MTTR value is not sufficient if CR nodes do not meet on multiple channels in a sequence period, since we cannot provide the reliable performance in CR ad hoc networks due to an easier link breakage caused by the appearance of PU signals. In the future work we will address underlying MAC protocol and will implement the proposed protocol on a CR platform.

Acknowledgments

The authors thank Professor Petri Mähönen for fruitful discussions. The authors thank the financial support from Deutsche Forschungsgemeinschaft and RWTH Aachen University through UMIC Research Centre. This work has benefited significantly from discussions with participants of European Union funded ACROPOLIS Network of Excellence (Grant ICT-257626).

References

[1] S. Haykin, *Fundamental Issues in Cognitive Radio*, Edited by S. Haykin, SpringerLink, 2007.

[2] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[3] S. Romaszko, D. Denkovski, V. Pavlovska, and L. Gavrilovska, "Asynchronous rendezvous protocol for cognitive radio ad hoc networks," in *Proceedings of the International Conference on Ad Hoc Networks (ADHOCNETS '12)*, October 2012.

[4] S. Romaszko and P. Mähönen, "Quorum systems towards an asynchronous communication in cognitive radio networks," *Hindawi Journal of Electrical and Computer Engineering*, vol. 2012, Article ID 753541, 22 pages, 2012.

[5] S. Lang and L. Mao, "A torus quorum protocol for distributed mutual exclusion," in *proceedings of the International Conference on Parallel and Distributed Systems (ICPADS '98)*, December 1998.

[6] J. M. Hall, *Combinatorial Theory*, John Wiley & Sons, 1986.

[7] W. S. Luk and T. T. Wong, "Two new quorum based algorithms for distributed mutual exclusion," in *Proceedings of the 17th International Conference on Distributed Computing Systems (ICDCS '97)*, pp. 100–106, May 1997.

[8] M. Singhal and N. G. Shivaratri, *Advanced Concepts in Operating Systems*, McGraw-Hill, New York, NY, USA, 1994.

[9] J. R. Jiang, Y. C. Tseng, C. S. Hsu, and T. H. Lai, "Quorum-based asynchronous power-saving protocols for IEEE 802. 11 ad hoc networks," in *Proceedings of the IEEE International Conference on Parallel Processing (ICPP '03)*, October 2003.

[10] Z. T. Chou, Y. H. Lin, and R. H. Jan, "Optimal fully adaptive power management protocols for asynchronous multi-hop ad hoc wireless networks," in *Proceedings of the 11th IEEE Singapore International Conference on Communication Systems (ICCS '08)*, pp. 569–573, November 2008.

[11] E. K. Lee, S. Y. Oh, and M. Gerla, "Randomized channel hopping scheme for anti-jamming communication," in *Proceedings of the IFIP Wireless Days (WD '10)*, October 2010.

[12] K. Bian, J. M. Park, and R. Chen, "A quorum-based framework for establishing control channels in dynamic spectrum access networks," in *Proceedings of the 15th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '09)*, pp. 25–36, September 2009.

[13] F. Hou, L. X. Cai, X. Shen, and J. Huang, "Asynchronous multichannel MAC design with difference-set-based hopping sequences," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 4, pp. 1728–1739, 2011.

[14] S. Romaszko and P. Mähönen, "Grid-based channel mapping in cognitive radio ad hoc networks," in *Proceedings of the 22nd Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '11)*, September 2011.

[15] S. Romaszko and P. Mähönen, "Torus quorum system and difference setbased rendezvous in cognitive radio ad hoc networks," in *Proceedings of the International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom '12)*, June 2012.

[16] W. Hu, D. Willkomm, M. Abusubaih et al., "Dynamic frequency hopping communities for efficient IEEE 802.22 operation," *IEEE Communications Magazine*, vol. 45, no. 5, pp. 80–87, 2007.

[17] H. Su and X. Zhang, "Channel-hopping based single transceiver MAC for cognitive radio networks," in *Proceedings of the 42nd Annual Conference on Information Sciences and Systems (CISS '08)*, pp. 197–202, March 2008.

[18] L. Jiao and F. Y. Li, "A single radio based channel datarate-aware parallel rendezvous MAC protocol for cognitive radio networks," in *Proceedings of the IEEE 34th Conference on Local Computer Networks (LCN '09)*, pp. 392–399, October 2009.

[19] S. Geirhofer, J. Z. Sun, L. Tong, and B. M. Sadler, "Cognitive frequency hopping based on interference prediction: theory and experimental results," *ACM SIGMOBILE Mobile Computing and Communications*, vol. 13, pp. 49–61, 2009.

[20] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Taxonomy and challenges of rendezvous algorithms in cognitive radio networks," in *Proceedings of the International Conference on Computing, Networking and Communications (ICNC '12)*, January2012.

[21] M. D. Silvius, F. Ge, A. Young, A. B. MacKenzie, and C. W. Bostian, "Smart radio: spectrum access for first responders," in *Wireless Sensing nand Processing III*, vol. 6980 of *Proceedings of the SPIE*, 2008.

[22] C. Cormio and K. R. Chowdhury, "Common control channel design for cognitive radio wireless ad hoc networks using adaptive frequency hopping," *Elsevier Ad Hoc Networks*, vol. 8, no. 4, pp. 430–438, 2010.

[23] C. Cormio and K. R. Chowdhury, "An adaptive multiple rendezvous control channel for cognitive radio wireless ad hoc networks," in *Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM WS '10)*, pp. 346–351, April 2010.

[24] N. C. Theis, R. W. Thomas, and L. A. DaSilva, "Rendezvous for cognitive radios," *IEEE Transactions on Mobile Computing*, vol. 10, pp. 216–227, 2010.

[25] H. Liu, Z. Lin, X. Chu, and Y. W. Leung, "Ring-walk based channelhopping algorithms with guaranteed rendezvous for cognitive radio networks," in *Proceedings of the International Workshop on Wireless Sensor, Actuator and Robot Networks (WiSARN-FALL '10), in conjunction with IEEE/ACM CPSCom*, China, December 2010.

[26] Z. Lin, H. Liu, X. Chu, and Y. W. Leung, "Jump-stay based channel-hopping algorithm with guaranteed rendezvous for cognitive radio networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 2444–2452, April 2011.

[27] C. M. Chao, H. C. Tsai, and K. J. Huang, "A new channel hopping MAC protocol for mobile ad hoc networks," in *Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP '09)*, November 2009.

[28] C. M. Chao and Y. Z. Wang, "A multiple rendezvous multi-channel MAC protocol for underwater sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '10)*, April 2010.

[29] M. Altamimi, K. Naik, and X. Shen, "Parallel link rendezvous in ad hoc cognitive radio networks," in *Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM '10)*, December 2010.

[30] K. Bian, J. M. Park, and R. Chen, "Control channel establishment in cognitive radio networks using channel hopping," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 689–703, 2011.

[31] Y. Zhang, Q. Li, G. Yu, and B. Wang, "ETCH: efficient Channel Hopping for communication rendezvous in dynamic spectrum access networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 2471–2479, chn, April 2011.

[32] K. Bian and J. M. Park, "Asynchronous channel hopping for establishing rendezvous in cognitive radio networks," in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM '11)*, pp. 236–240, April 2011.

[33] Y. S. Hsieh, C. W. Lien, and C. T. Chou, "A multi-channel testbed for dynamic spectrum access (DSA) networks," in *Proceedings of the 7th ACM workshop on Wireless multimedia networking and computing (WMUNEP '11)*, Miami, Fla, USA, October 2011.

[34] S. Romaszko and P. Mähönen, "Quorum-based channel allocation with asymmetric channel view in cognitive radio networks," in *Proceedings of the 6th ACM Performance Monitoring, Measurement and Evaluation of Heterogeneous Wireless and Wired Networks Workshop (PM2HW2N '11)*, October 2011.

[35] D. R. Sinson, *Combinatorial Designs: Constructions and Analysis*, Springer, 2004.

[36] L. A. DaSilva and I. Guerreiro, "Sequence-based rendezvous for dynamic spectrum access," in *Proceedings of the 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '08)*, pp. 440–446, October 2008.

[37] Y. R. Kondareddy, P. Agrawal, and K. Sivalingam, "Cognitive radio network setup without a common control channel," in *Proceedings of the IEEE Military Communications Conference (MILCOM '08)*, November 2008.

[38] R. Zheng, J. C. Hou, and L. Sha, "Asynchronous wakeup for ad hoc networks," in *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '03)*, pp. 35–45, June 2003.

[39] J. R. Jiang, Y. C. Tseng, C. S. Hsu, and T. H. Lai, "Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks," *Mobile Networks and Applications*, vol. 10, no. 1, pp. 169–181, 2005.

[40] S. Romaszko, "Making a blind date the guaranteed rendezvous in cognitive radio ad hoc networks," in *Proceedings of the 18th European Wireless Conference (EW '12)*, April 2012.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 415182, 8 pages
doi:10.1155/2012/415182

Research Article

Multidomain Hierarchical Resource Allocation for Grid Applications

Mohamed Abouelela and Mohamed El-Darieby

Software Systems Engineering Department, University of Regina, Regina, SK, Canada S4S 0A2

Correspondence should be addressed to Mohamed Abouelela, mmostafa79@gmail.com

Received 4 May 2012; Revised 4 August 2012; Accepted 5 August 2012

Academic Editor: Fangwen Fu

Geographically distributed applications in grid computing environments are becoming more and more resource intensive. Many applications require the collaboration between different domains, may be independently administrated domains, to exchange data and share computing and storage resources. This collaboration should be done in a way that maintains the privacy of each participant domain. This calls for new architectures and approaches to deal with such multidomain environments. We propose a hierarchical-based architecture as well as multidomain hierarchical resource allocation approach. The resource allocation is performed in a distributed way among different domains such that each participant domain keeps its internal topology and private data hidden while sharing abstracted information with other domains. Both computing and networking resources are jointly scheduled while optimizing the application completion time taking into account data transfer delays. Simulation results show the scalability and feasibility of the proposed approach.

1. Introduction

An increasing number of scientific and enterprise applications are becoming dependent on high performance computing (HPC) environments. In general, these applications are computation- and communication-intensive as they process very large amounts of datasets. The datasets of the applications and the resources required are geographically distributed across the grid.

The grid is an interconnected multidomain environment where each domain consists of computational, storage, and communication resources grouped together for business or administrative reasons. Each domain is independently administrated and is free to deploy different technologies. Meeting resource requirements of HPC applications generally requires allocating resources across a number of grid domains without sacrificing domain security or privacy requirements. This calls for novel multidomain scalable and reliable grid architectures, mechanisms, and algorithms that keep the balance between integration and privacy.

In general, grid systems should maintain scalability, reliability, domain privacy, and integration requirements. a scalable grid system implies maintaining acceptable performance as the number of domains increase and as the workload on the system intensifies. Reliability implies the ability of the architecture to recover from resource failures in acceptable time. Grid resource integration is a basic concept in grid computing systems that results in better overall system performance and resource utilization. The privacy of a grid domain must be maintained in for confidentiality and commercial competition.

In this paper, we propose hierarchical-based architecture. Hierarchical architecture is typically used to handle scalability and privacy problems [1]. The proposed hierarchical architecture helps in keeping domain privacy while integrating with other domains. For each domain, different computing and networking resource parameters including internal topology and resource status information are kept internally, while abstracted values for these parameters are shared with other domains. The abstracted values are to be sent to a higher level resource manager to help in taking the resource allocation decisions at the interdomain level. A multidomain hierarchical resource allocation approach is used for resource allocation.

The multidomain hierarchical allocation approach is carried out in a distributed manner. Each domain executes intradomain coallocation algorithms to allocate its own resources. Moreover, different domains coordinate with each other for resource allocation at the interdomain level and over all hierarchical levels. The approach relies on coallocation algorithm that jointly allocate computing and networking resources considering both data execution and data transfer times. We focus on a computation- and communication-summarized in section intensive application where data is stored at different sites across multidomain network and can be divided into independent subsets to be processed in parallel at different locations. This type of application is called *Divisible Load application*.

The rest of the paper is organized as follows: related work is summarized in Section 2. The proposed architecture is described in Section 3, while the multidomain hierarchical resource allocation approach is explained in Section 4. Experiments setup is explained in Section 5, while results and discussions are provided in Section 6. Finally, conclusions are offered at the end of the paper.

2. Related Work

Resource allocation in high performance grid computing is an area of ongoing research and development. Many researches were conducted illustrating the joint allocation approach and showing the advantage of it over the separated one [2–6]. Most of these efforts assume a centralized resource manager that has a complete vision of network topology as well as networking and computing resources status. This assumption is not valid for large-scale worldwide grid networks. Practically, grid network comprises geographically distributed heterogeneous resources interconnected by multidomains networks. Each domain is managed by a local domain grid manager that is usually not willing to share its internal domain information to others due to security and business confidentiality reasons. Moreover, maintaining and managing, in one centralized location, dynamic data coming from heterogeneous resources located in multidomain environments added a serious difficulty to the resource allocation process. To deal with such multidomain environments, two solutions were presented in the literature: network virtualization [5, 7, 8] and harmony [9].

Network virtualization separates logical network, called virtual network, from the substrate infrastructure network resources by dividing the role of the traditional service provider into two independent entities: infrastructure provider, who manages the substrate infrastructure network resources and service provider, who creates the virtual network by aggregating network resources from multiple infrastructure providers to build the network topology. A number of projects were already developed providing *network virtualization* over multiple domains [5, 8].

Using *network virtualization* as a solution of multidomain joint scheduling problem in grid computing environment is proposed in [5]. The authors proposed a virtualized optical network (VON) service composition framework for grid applications. Upon application task arrival, the virtual network topology is generated, and the joint scheduling starts over the virtualized network. Then, the virtualized network is release after finishing the task. Using *network virtualization* as described in [5] has many drawbacks. *Network virtualization* is still an evolving technique facing many challenges and enclosing lots of complexities [7]. Integrating multidomain joint scheduling problem within the *network virtualization* framework increases the complexity of the system without any clear advantage. One of the major drawbacks is the proposed virtualized network topology design. The proposed topology design uses bounds for the maximum amount of the expected traffic to calculate the minimum bandwidth to be reserved. This topology design does not take into account the availability of computational resources. It is not acceptable to consider just the expected traffic bounds while designing a topology that will be used in joint computing and networking resource scheduling. Ignoring computational resources capacity and availability may affect the overall performance significantly specially in computational intensive applications.

Harmony [9] is the network resource brokering system in Phosphorus Research Project [10]. The objective of Phosphorus project is to provide on-demand and end-to-end provisioning of computing and networking resources in multidomain and multitechnology environments. The workflow when a grid task received is as follows. After the authentication, the availability of the requested resources is verified. Then, the end-to-end path is allocated in two phases. In the first phase, the interdomain path is selected by the interdomain broker (IDB) module. In the second phase, a Network Resource Provisioning System (NRPS) module in each independent domain calculates the intradomain path. The intradomain topology for each domain is totally hidden from other domains and from IDB. Only border endpoints and interdomain links are exported. The organization of IDBs can be done in centralized, hierarchical, and distributed manner.

The proposed Harmony system for multidomain reservation is promising. It shows how different domains can interact to provide end-to-end connectivity and allocate the required networking resources, while maintaining the confidentiality for each domain. The main concern of this work is the allocation of the networking resource in multidomain environment, while the joint allocation of the computing and networking resources is not presented. It is just stated that they assumed that the computing resources are scheduled prior to path setup request.

In this paper, we extended the hierarchical architecture of the Harmony system to jointly schedule both computing and networking resources in multidomain environment. Each domain will maintain its structure and topology internally, while share an abstracted data about its computing and networking resources status with its Resource Manager (RM). The RM is similar to IDB in Harmony system with extended functionality to manage both computing and networking resources. The RMs are to be arranged in a multilevel hierarchical architecture. New approaches to

schedule divisible load applications in such multidomain hierarchical architecture are to be introduced.

Scheduling divisible load applications in distributed environments is frequently discussed in the literature. divisible load theory (DLT) has been successfully applied to parallel and distributed systems, as well as to grid computing environment [11–13]. Genetic-algorithms (GA)-based approaches were also proposed to schedule Divisible Loads [2, 14]. Integer linear programming has also been introduced to model such problems [13].

3. Proposed Architecture

3.1. Problem Statement. HPC grid computing applications require heterogeneous and geographically distributed computing resources interconnected by multidomain networks. Cooperation among domains, without sacrificing domain privacy, to allocate resources is required to execute such applications. For example, internal topology information of a domain should not be revealed to other domains [9]. This calls for a novel and scalable architecture allowing the integration between domains while keeping the privacy of each domain is required. We focus on divisible load applications in multidomain environment where application data is originally stored in geographically distributed sites and is divided into independent subsets to be executed in parallel at distributed data-processing sites. Those sites belong to different independently administrated domains. The performance of these applications can be optimized by concurrent execution of data processing tasks at different processing sites with different input datasets.

Such applications are modelled as data processing jobs requiring large logical input dataset, D, of total size L. D is divided into n physical datasets stored at different data sources DS_k, where $k = 1, \ldots, n$. Each physical dataset k has a size L_k, where $\sum_{k=1}^{n} L_k = L$. Those datasets are to be divided into n datasets to be executed at n different sites, and assign the required computing and networking resources. We assume that divisible data can be executed at any site using the same data processing algorithm.

The optimization (scheduling) problem is to minimize the maximum completion time by deciding on portions of datasets to be executed at each site (either executed at sites belonging to the same domain or different domains) and assigning necessary inter- and intradomain computing and networking resources.

3.2. Hierarchical Architecture. Hierarchical architecture is typically used to handle scalability and privacy problems [1]. In hierarchical architecture, sites with storage and computing resources are organized into different interconnected subnetworks (domains). A domain consists of a number of interconnected sites. A RM manages and maintains topological and state information about different computing and networking resources in a domain. The process of grouping sites (at one hierarchy level) into logical domains and abstracting such domains via a RM (at the next higher level) is done at all levels of the hierarchy (see Figure 1).

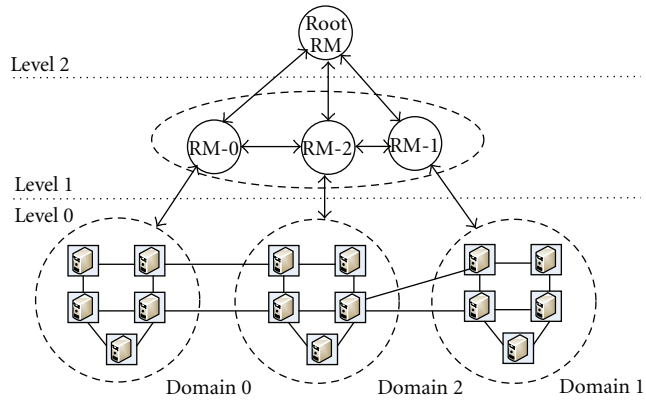


Figure 1: Two levels of hierarchical grid architecture.

Figure 1 shows a screenshot for a two levels hierarchical grid architecture. Level-0 nodes (square shape) represent data-processing sites, for example, computing clusters, or super computers containing storage and processing capabilities. Different sites and links have different computational and networking capabilities. Different sites are grouped into domains. This does not violate the special case by which single site can be considered as a domain. Each domain is managed by a level-1 RM (circular shape). Vertical line represents dedicated control channels between level-1 RM and the corresponding domain sites. Level-1 RMs are grouped into domains. Level-1 domains are managed by Level-2 RM which aggregates the collected information by level-1 RMs. In this two levels example, Level-2 RM is known as the root RM.

Level-0 horizontal links presents inter- and intradomain links available for data transfers. Level-1 horizontal links connecting level-1 RMs are virtual links. Virtual links represent the aggregated topology of the corresponding level-0 interdomain links, by which Level-0 interdomain links are aggregated and represented by level-1 links. The capacity of level-1 virtual link is the summation of the capacities of the corresponding level-0 interdomain links. At each level in the hierarchy, networking and computing resource status information is aggregated by the corresponding RM, abstracted and sent to parent RM. Within this architecture, we assume the following.

(i) RMs are connected to each other and to physical sites with fault tolerant connections (control channels).

(ii) Due to privacy considerations, complete data for each site, including its internal topology and static and dynamic resource status data, is available only for its domain RM.

(iii) An RM shares border end points and interdomain links data for its managed domain with its parent RM.

(iv) An RM maintains complete vision for the sites connected directly to it, and summarized vision (abstracted parameters) of the sites managed by its children RMs.

4. Multidomain Hierarchical Resource Allocation Approach

The resource allocation is carried out in a distributed manner at RMs from different domain and different hierarchical levels. The first step starts by executing resource coallocation algorithm at the root RM with the objective of achieving load balance and minimizing the application completion time. The algorithm defines interdomain data transfer requests. In the following step, the data transfer requests are sent down the hierarchy to children RMs. Children RMs apply intradomain resource allocation algorithm to allocate their own resources independently. This step is repeated down the hierarchy until the data transfer requests reaches Level-0 sites. If a RM couldont find enough resources to fulfil the requests, a relocate message is to be sent up the hierarchy to its parent RM to relocate the request load to another RM.

The process starts, as the system receives a job request for divisible load application, at the root RM, assumed at level k. The RootRM calculates the level $k-1$ RMs that have enough resources to meet application request. The rootRM defines a list of data transfer requests for each of the level $k-1$ RM that is expected to participate in serving the application request. Each *data transfer request* is defined by five components: source, destination, value-to-transfer, path and start-time. The *source* is a node with a number of datasets (equals to the value-to-transfer) to be executed remotely at the destination. Those datasets should be sent at a certain time (start-time) and should follow a certain *path*. The *path* is defined as a number of links connecting *source* and *destination* nodes. At level $k-1$, the RMs schedule their resources according to the requests by their parent RM. This process is repeated at each level in the hierarchy until Level 0 sites receives the resource allocation requests. This completes the scheduling process.

For example, consider the grid architecture introduced in Figure 1. The resource allocation is done first at root RM, which defines a list of data transfer requests. Assume that one of the defined requests is source = RM-0, destination = RM-1, start time = 40 s, value-to-transfer = 20 datasets, and path = RM-0 ⇒ RM-2 ⇒ RM-1 (the interdomain path). This request is to be sent to all the RMs involved in this task (RM-0, RM-1, and RM-2). Then, the scheduling starts at those RMs to allocate the required internal resources to complete those requests and provide end to end connectivity. The scheduling at each of the three RMs results in new lists of requests. Those lists are to be sent to the sites at level 0.

The detailed algorithm at each RM is comprised of the following three steps, described in the following subsections.

4.1. Handling Parent Data Transfer Requests. The process at a level j RM starts by receiving a list of data transfer requests from its parent RM at level $j+1$. Those requests should be handled first by allocating the necessary computing and networking resources. A resource allocation greedy algorithm is called to allocate the needed resources. This greedy algorithm will be explained in Section 4.3. The RM defines the set of need computing and networking resources according to its role in parent request. Generally, the RM can play one of the following roles: *source role*, *destination role,* and *transit role*.

(i) *Source Role*. If the data transfer request defines the RM as a source node, then a number of datasets, equals to the *value-to-transfer*, should be sent out of the domain managed by this RM to a Predefined interdomain link at, or before, task start-time. The RM should allocate the required internal networking resources to transfer the task data to the border node connected to the predefined interdomain link.

(ii) *Destination Role*. If the data transfer request defines the RM as a destination node, then the domain managed by this RM expects a certain number of datasets to arrive to a certain border node through a certain interdomain link. The required computing and networking resources should be assigned to execute or analyze the coming data internally.

(iii) *Transit Role*. If the data transfer request defines the RM as a transit node (one of the intermediate nodes defined in the path field of the request), then the managed domain expects a certain amount of data to arrive to a certain border node and the same amount of data to send out from another border node. The RM should provide the internal networking resources to connect those two border nodes to complete the interdomain path end to end connectivity.

4.2. Optimal Load Distribution Calculation. Allocating the necessary resources to handle parent request may result in unbalanced-load distribution among different computing units. Therefore, load balancing is needed to ensure that the computing units in the participant sites (or domains) will finish the load processing at the same time. Assuming n sites, L_i, for all $i \in 1,\ldots,n$ defines the current load distribution (before load balancing). The objective of the load balancing is to define the optimal load distribution α_i, for all $i \in 1,\ldots,n$. α_i defines the number of datasets that should be allocated for each site i for optimal load distribution. Different algorithms could be used to calculate the optimal load distribution. in this paper, we will use Network Aware Divisible Load Algorithm (NADLA) [15]. NADLA is a simple, light-weight and fast load balancing algorithm based on divisible load theory. It considers network availability and connectivity while deciding on load distribution.

4.3. Resource Allocation Greedy Algorithm at Each RM. After defining the optimal load distribution, a set of *data transfer requests* should be defined to execute the new load distribution, and different computing and networking resources should be allocated. The resource allocation greedy algorithm (Algorithm 1) is used to define the requestlist (the set of data transfer requests). The algorithm starts with an empty requestlist (step 1). Then, the difference between the optimal load distribution α_i and the current load distribution L_i is calculated for each site i. This difference represents the portions of data to be transferred to/from each site i. This value can be positive, negative, or zero. Positive

1: Set $RequestList = \{\}$
2: Calculate $\alpha_i - L_i, \forall i \in 1, \ldots, n$
3: **while** $\alpha_i - L_i \neq 0, \forall i \in 1, \ldots, n$ **do**
4 : Set $dest = i$, such that $(\alpha_i - L_i)$ is max
5 : Set $SourceList = \{i\}, \forall i \in 1, \ldots, n \& \alpha_i - L_i \leq 0$
6 : **for each** $source \in SourceList$ **do**
7 : Calculate $Path_{source,dest}$ and $PathWaitingTime_{source,dest}$
8 : **end for**
9 : Select $source$ from $SourceList$ such that $PathWaitingTime_{source,dest}$ is minimum
10 : Set $ValueToTransfer = min[abs(\alpha_{source} - L_{source}), abs(\alpha_{dest} - L_{dest})]$
11 : $RequestList+ = newTask(source, dest, ValueToTrasfer, Path_{source,dest}, TransferTime)$
12 : Set $L_{source} - = ValueToTransfer$
13 : Set $L_{dest} + = ValueToTransfer$
14 : Update links with the new reservations
15: **end while**
16: Populate $RequestList$

Algorithm 1: Resource allocation greedy algorithm.

values mean data sink site (destination receiving data to be executed internally), while negative values mean data source sites (sites containing extra-data to be executed in remote site).

The algorithm iterates until the current load distribution equals to the optimal load distribution at all sites. In each iteration, the *dest* site is selected first as the less loaded site; $\alpha_i - L_i$ is maximum. Then, a *SourceList* list is defined containing all sites having extra-load; $\alpha_i - L_i$ is negative. The *source* site is selected from this list with the objective of minimizing the path waiting time. The shortest paths between the *dest* site and each site in the *SourceList* are calculated, and the site with minimum path waiting time is selected. A new *data transfer request* is added to the *RequestList*. Finally, the source and destination Loads (L_{source} and L_{dest}) and Links schedules should be updated accordingly.

5. Experiment Setup

Simulation experiments were conducted to evaluate the performance of proposed architecture as well as the multidomain resource allocation approach. A wide range of different parameters was considered to cover different network topologies, application types, and algorithms. Up to 10 runs are carried out for each experiment and their results are averaged for 95% confidence intervals.

Simulations were conducted using OMNET++ network simulator (http://www.omnetpp.org/). OMNET++ is a C++ open source discrete event simulator. OMNET++ is highly modular and well-structured simulator. It provides realistic and accurate network models for different protocols and architectures. We developed our own modules to support multilevel hierarchical architecture and grid computing functionality.

Different network topologies were generated with a wide range of parameter variations matching the network architecture proposed in Section 3.2. Different network sizes, the number of level 0 sites, were considered, varying from 16 sites up to 1000 sites. Sites were grouped into domains to construct multilevel hierarchies up to 5 levels. Networks with different average node degree d: the number of links connecting this node to other nodes, were considered. The average node degree values are varying from 2 to 8. Different bandwidth values for interdomain and intradomain links were considered.

Moreover, different application load sizes were examined starting from an average of 25 datasets per source site to 3000 datasets per site, while the unit dataset size was fixed to be 1 Gbit. As the datasets per source site increases, the application becomes more data-intensive. Different applications may have different processing capacities (time to process a unit dataset) even on the same site. As the processing capacity increases, the application goes to be more computationally intensive. In our simulations, we considered three application categories: data intensive applications, intermediate applications, and computationally intensive applications. To differentiate between those three categories, the average site processing capacities is set to 5, 25, and 100 second/unit dataset for the three categories, respectively.

The following metrics have been used to evaluate the performance of the hierarchical scheduling as well as the proposed architecture.

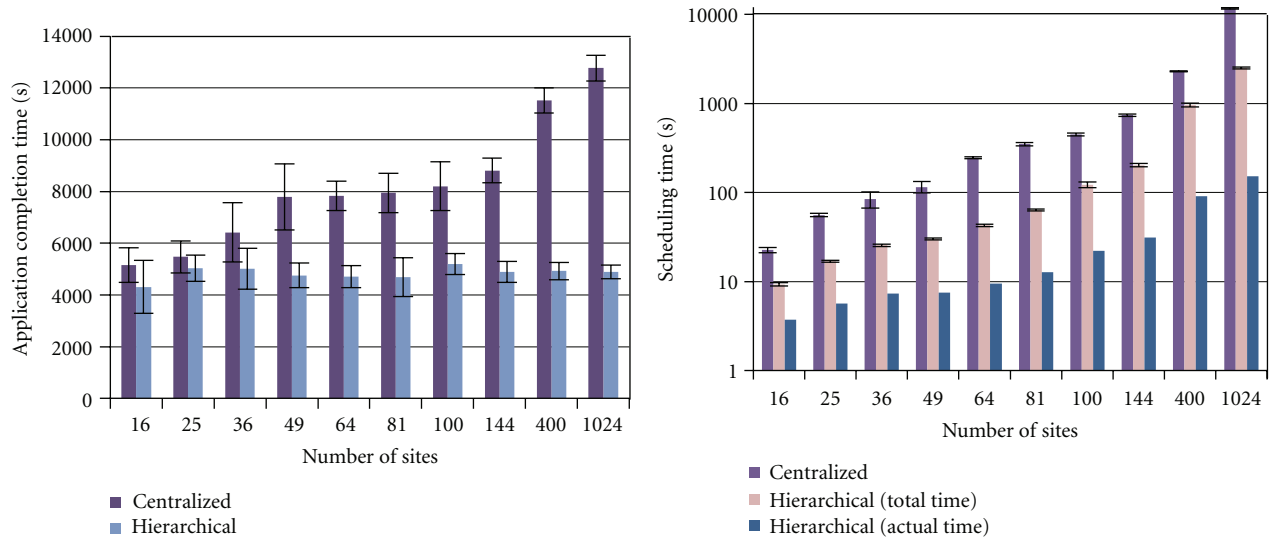
(i) *The application completion time*, which is the maximum task completion time over all the sites. It is measured from the task arrival time, until the last site finishes data processing.

(ii) *Scheduling time*, which is the time consumed inside the RMs to come to a decision on scheduling and resource allocation. For hierarchical scheduling, the scheduling time is calculated as the summation of scheduling times over all RMs.

(iii) *Standard Deviation in resource utilization* (SD), which is a metric of the system load balancing by measuring the variations in resource utilization. It is calculated as the standard deviation in resource utilization for both links and computing resource units. Resource utilization is calculated as the percentage of time by which the resource is busy, so the SD is calculated as a percentage. SD = 0% reflects optimal load balancing, that all the resources are utilized equally.

6. Results and Discussion

6.1. Hierarchical versus Centralized. The hierarchical and centralized architectures are compared for different network sizes varying from 16 sites network up to 1024 sites network. The number of hierarchical levels is fixed to two levels for all hierarchical networks; also the average node degree is fixed to 4.



(a) Application completion time

(b) Scheduling time

Figure 2: Hierarchical versus centralized for different network sizes.

Figure 2(a) shows the application completion time for both centralized and hierarchical architectures. It is clear that for small size networks (16,25 & 36 sites) the two architectures results in almost the same application completion time. As increasing the network size the centralized architecture results in significant increase in the application completion time, while the hierarchical architecture results in slight increase in the application completion time.

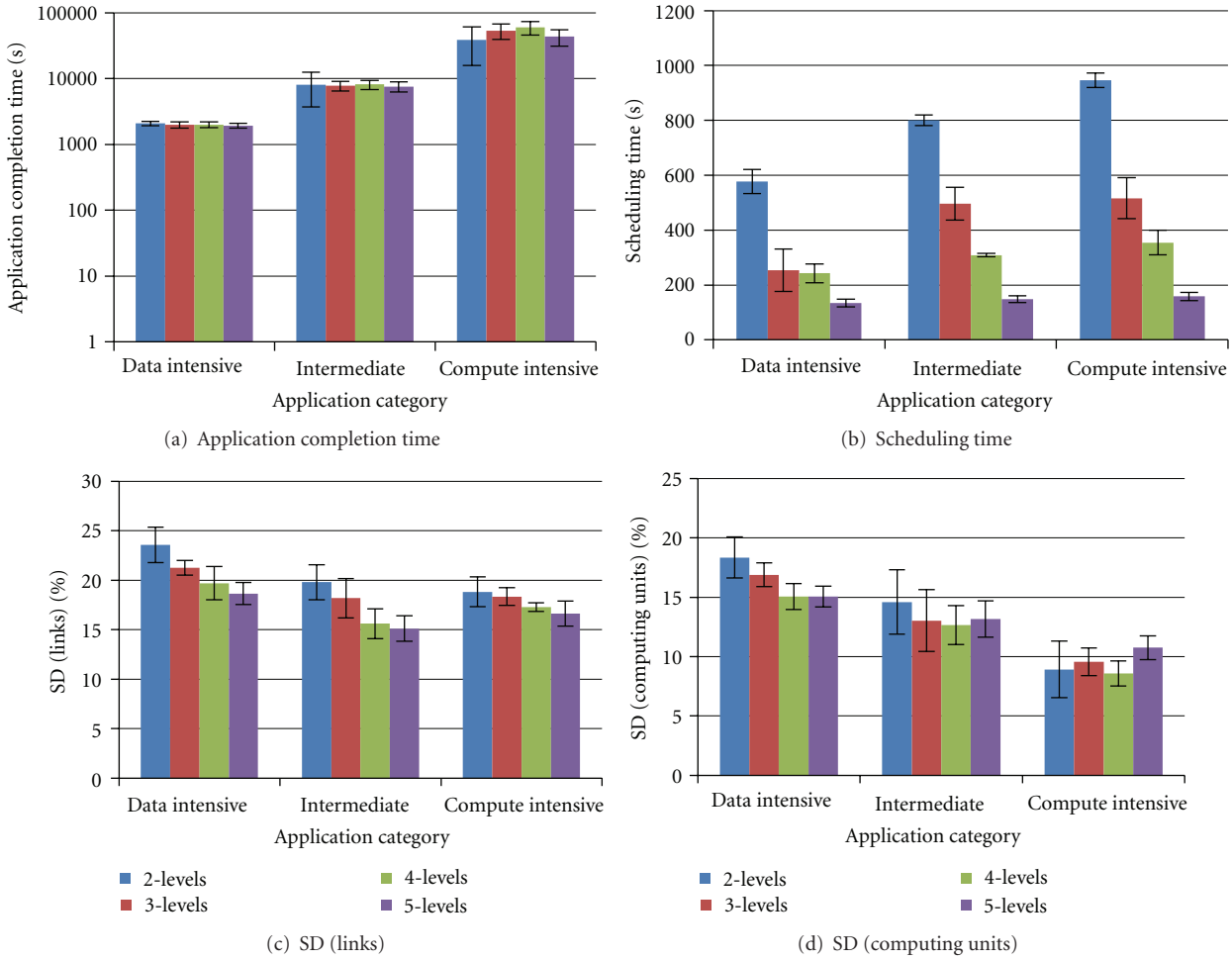
The *scheduling time* for both architectures, shown on the *y*-axis in Figure 2(b), (log scale is used for the *y*-axis for better visualization of results). For hierarchical architecture, two values for the scheduling time were measured: the total *scheduling time* and the *actual scheduling time*. The total time is measured as the summation of the scheduling task times in all the RMs, while the actual time is the actual task scheduling time by considering that the RMs at the same level are running in parallel. This is one of the advantages of the hierarchical architecture, that the scheduling is distributed among a number of RMs running in parallel, which results in significant deduction in scheduling time. As shown in Figure 2(b), the hierarchical architecture outperforms the centralized one for the *scheduling time*.

A significant reduction in the *scheduling time* is measured when using the hierarchical one especially for large size networks when compared to centralized architectures. The reduction is around 90% for small size networks (16 & 25 sites networks), while it reaches around 98% for large size networks (1024 sites network). This comes at a specific cost that will be discussed below. The *scheduling time* for centralized architecture increases significantly as increasing the network size. For example, as increasing the number of sites from 25 to 49, the *scheduling time* increases by a factor of 2, while increasing the number of sites from 400 to 1024 results in increase by a factor of 5. On the other hand, for hierarchical architecture, the *scheduling time* increases by a factor of 1.65 while increasing the number of sites from 400 to 1024.

6.2. Effect of Hierarchy Depth. To evaluate the effect of the depth of hierarchy, different networks with fixed number of sites (250 site) were used while varying the number of hierarchical levels from 2 to 5. The depth of the hierarchy is evaluated for the three application categories and the results are shown in Figures 3(a), 3(b), 3(c), and 3(d). It is shown in Figure 3(a) that *application completion time* is not affected by increasing the depth of the hierarchy for networks with different hierarchical levels. Figure 3(b) shows a decrease in the *scheduling time* as increasing the depth of the hierarchy for the three application categories. In addition, a small decrease in the standard deviation *in links utilization* is reported (Figure 3(c)). This means better load balancing among different links as increasing the depth of the hierarchy. Figure 3(d) shows no notable change in the *standard deviation in computing units utilization* as increasing the depth of the hierarchy.

Those advantages in *scheduling time* could be verified analytically. The time of the scheduling algorithm executed at each RM depends mainly on the number of sites in the managed domain. Assuming that we have a total of N sites grouped into domains in L hierarchical levels. Then, the number of nodes in each domain follows the exponential function $N^{1/L}$. Then, the scheduling time at each RM will follow the same function and decrease exponentially with respect to the number of levels. Assuming that all the RMs at the same level are running in parallel, then the actual scheduling time equals to the result of multiplying the scheduling time at one RM by the number of levels.

Those advantages in the *scheduling time* as using the hierarchical architecture as increasing the depth of the hierarchy come at the cost of increasing the control overhead. Increasing the depth of the hierarchy increases the required number of RMs to manage the system for networks with the same size. For example, increasing the depth of the hierarchy from 2 to 5 increases the number of RMs by



(a) Application completion time

(b) Scheduling time

(c) SD (links)

(d) SD (computing units)

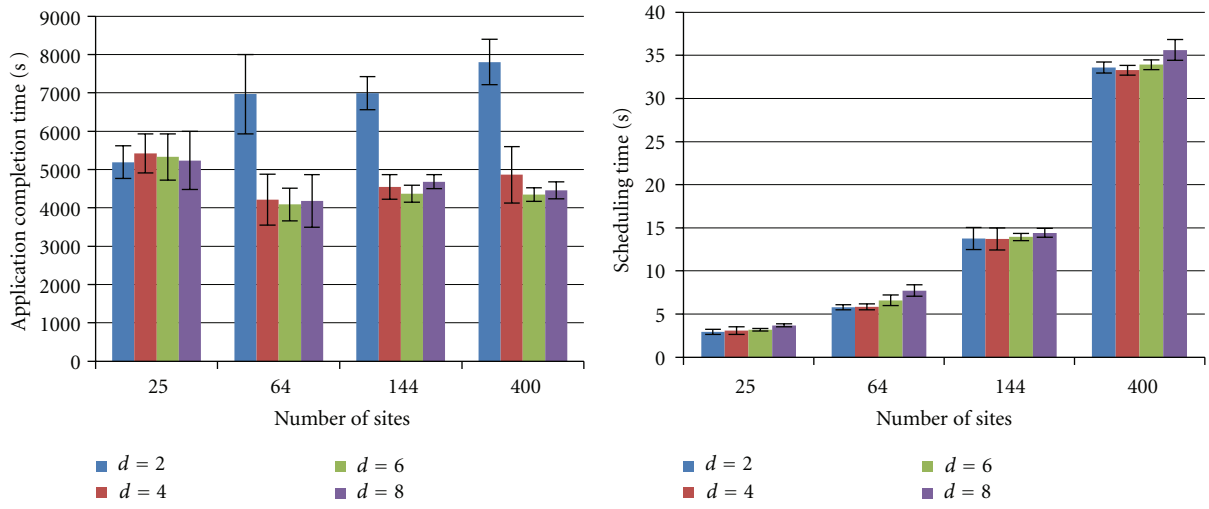
Figure 3: The Effect of depth of hierarchy for different application categories.

a factor of 7. Controlling and maintaining this hierarchical structure increases the cost and complexity as increasing the number of RMs. This increases the control overhead and communication complexity. In [1], a study of a hierarchical routing protocol reported a notable increase in path setup time and communication overhead as increasing the depth of the hierarchy.

6.3. Effect of Node Degree. To study the effect of average node degree (d), networks with different sizes and average node degrees were considered. As shown in Figure 4(a), for small size network (25 sites network), no significant change in *application completion time* is reported as increasing the average node degree, while for larger size networks (64, 144 & 400 sites networks), a notable deduction in *application completion time* is reported while increasing the average node degree from 2 to 4. Increasing the average node degree more than 4 results in no notable improvement in the application completion time for all the tested network sizes. Figure 4(b) shows a very small increase in *scheduling time* as increasing the average node degree. Those results can help in network dimensioning problem to select the optimal value for the average node degree that reduces the application completion time, while minimizing the number of links. The dimensioning problem is out of scope of this paper and may be considered in the future.

7. Concluding Remarks

The proposed hierarchical architecture as well as the multidomain hierarchical resource allocation approach provided a novel solution for the joint resource allocation problem in multidomain grid environments. The hierarchical architecture maintained the scalability and privacy of the grid system. The proposed architecture helped in keeping the domain privacy while integrating with other domains. Domain internal topology and resource status information were kept internally, while sharing abstracted parameters with other domains. The multidomain hierarchical resource allocation approach was carried out in a distributed manner where each domain executes intradomain joint scheduling algorithm to schedule its own resources. Moreover, the process involved coordinating the resource allocation at the interdomain level and over all hierarchical levels.



(a) Application completion time

(b) Scheduling time

Figure 4: Effect of average node degree for different network sizes.

Simulations were conducted to evaluate the performance in terms of application completion time, scheduling time, and resource utilization for different network topologies, application types, and algorithms. The proposed hierarchical architecture proved its scalability and feasibility. Increasing the hierarchical depth results in better scheduling time and load balancing. Those advantages came at the cost of increasing control overhead. In the future, other research work will be conducted based on the proposed hierarchical architecture and hierarchical resource allocation approach. Analysing and evaluating different aggregation procedures is one of our future goals. In addition, introducing fault management mechanism will be considered.

References

[1] M. El-Darieby, D. Petriu, and J. Rolia, "Load-balancing data traffic among inter-domain links," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 5, pp. 1022–1033, 2007.

[2] M. Abouelela and M. El-Darieby, "Co-scheduling computational and networking resources in E-science optical grids," in *Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM '10)*, pp. 1–5, Miami, FLa, USA, December 2010.

[3] N. Charbonneau, V. M. Vokkarane, C. Guok, and I. Monga, "Advance reservation frameworks in hybrid IP-WDM networks," *IEEE Communications Magazine*, vol. 49, no. 5, pp. 132–139, 2011.

[4] M. Koseoglu and E. Karasan, "Joint resource and network scheduling with adaptive offset determination for optical burst switched grids," *Future Generation Computer Systems*, vol. 26, no. 4, pp. 576–589, 2010.

[5] Y. Wang, Y. Jin, W. Guo, W. Sun, and W. Hu, "Virtualized optical network services across multiple domains for grid applications," *IEEE Communications Magazine*, vol. 49, no. 5, pp. 92–101, 2011.

[6] G. Zervas, E. Escalona, R. Nejabati et al., "Phosphorus grid-enabled GMPLS control plane (G2MPLS): architectures, services, and interfaces," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 128–137, 2008.

[7] N. M. M. K. Chowdhury and R. Boutaba, "Network virtualization: state of the art and research challenges," *IEEE Communications Magazine*, vol. 47, no. 7, pp. 20–26, 2009.

[8] I. Houidi, W. Louati, W. Ben Ameur, and D. Zeghlache, "Virtual network provisioning across multiple substrate networks," *Computer Networks*, vol. 55, no. 4, pp. 1011–1023, 2011.

[9] A. Willner, C. Barz, J. A. Garcia Espin, J. Ferrer Riera, S. Figuerola, and P. Martini, "Harmony—advance reservations in heterogeneous multidomain environments," in *Proceedings of the 8th International IFIP-TC 6 Networking Conference (NETWORKING '09)*, pp. 871–882, Springer, Berlin, Germany, 2009.

[10] S. Figuerola, N. Ciulli, M. De Leenheer, Y. Demchenko, W. Ziegler, and A. Binczewski, "Phosphorus: single-step on-demand services across multi-domain networks for e-science," in *Proceedings of the Network Architectures, Management, and Applications V (SPIE '07)*, vol. 6784, Wuhan, China, November 2007.

[11] S. Viswanathan, B. Veeravalli, and T. G. Robertazzi, "Resource-aware distributed scheduling strategies for large-scale computational cluster/grid systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1450–1461, 2007.

[12] C. Yu and D. C. Marinescu, "Algorithms for divisible load scheduling of data-intensive applications," *Journal of Grid Computing*, vol. 8, no. 1, pp. 133–155, 2010.

[13] P. Thysebaert, B. Volckaert, M. De Leenheer, F. De Turck, B. Dhoedt, and P. Demeester, "Dimensioning and on-line scheduling in Lambda Grids using divisible load concepts," *The Journal of Supercomputing*, vol. 42, no. 1, pp. 59–82, 2007.

[14] S. Kim and J. B. Weissman, "A genetic algorithm based approach for scheduling decomposable Data Grid applications," in *Proceedings of the International Conference on Parallel Processing (ICPP '04)*, pp. 406–413, August 2004.

[15] M. Abouelela and M. El-Darieby, "Towards network-aware divisible load theory for optical grids," in *Proceedings of the IEEE 13th International Conference on High Performance Computing and Communications (HPCC '11)*, pp. 425–431, September 2011.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 358647, 17 pages
doi:10.1155/2012/358647

Research Article

Virtual Network Embedding: A Hybrid Vertex Mapping Solution for Dynamic Resource Allocation

Adil Razzaq, Markus Hidell, and Peter Sjödin

School of ICT, KTH Royal Institute of Technology, 16440 Kista, Sweden

Correspondence should be addressed to Adil Razzaq, arazzaq@kth.se

Received 2 March 2012; Revised 14 May 2012; Accepted 16 May 2012

Academic Editor: Shuo Guo

Virtual network embedding (VNE) is a key area in network virtualization, and the overall purpose of VNE is to map virtual networks onto an underlying physical network referred to as a substrate. Typically, the virtual networks have certain demands, such as resource requirements, that need to be satisfied by the mapping process. A virtual network (VN) can be described in terms of vertices (nodes) and edges (links) with certain resource requirements, and, to embed a VN, substrate resources are assigned to these vertices and edges. Substrate networks have finite resources and utilizing them efficiently is an important objective for a VNE method. This paper analyzes two existing vertex mapping approaches—one which only considers if enough node resources are available for the current VN mapping and one which considers to what degree a node already is utilized by existing VN embeddings before doing the vertex mapping. The paper also proposes a new vertex mapping approach which minimizes complete exhaustion of substrate nodes while still providing good overall resource utilization. Experimental results are presented to show under what circumstances the proposed vertex mapping approach can provide superior VN embedding properties compared to the other approaches.

1. Introduction

Internet is being utilized to provide a wide range of services. Over a period of time (which is not too long), it has become vital component/core architecture to provide services for global commerce, media, and defense [1].

In spite of the success attributed with the current Internet, it has some flaws which need to be addressed. The "everything over IP" [2], as well as "best-effort" packet delivery does not suit all the services being provided on current Internet, whereas security, routing stability, and control and QoS (quality of service) guarantees are also some of the major concerns [1].

However, there are many limitations/obstacles in overcoming the above mentioned flaws. Some of these include appropriate changes in routers and host software, as well as joint agreement of all the ISPs on any architectural change [3]. Capital investment, competing interests of stakeholders as well as end-to-end design of IP, calls for a worldwide agreement to introduce any changes [1]. Since, it is very rare that a single ISP controls complete end-to-end path, new services have only been employed/tested within small geographic locations [4].

The challenges/requirements to overcome the Internet impasse/ossification were outlined in [3–5]. Requirements mentioned in [3] include, ease of experimentation with new architectures on live traffic, provisioning of a plausible deployment path for an architecture, and focusing of an architectural solution on a broad range of problems. The challenges described in [4] are, discovering the resources of a physical infrastructure, assigning virtual networks to underlying physical networks, and accounting of resources. Isolation, performance, scalability, flexibility, evolvability, management, and applications were the challenges identified for new generation network architectures (future network) in [5].

Network virtualization is at the heart of proposals for addressing the Internet ossification [1, 3, 4]. It can be utilized in experimental research facilities [6–8] as well as in

provision of customized end-to-end services over a shared infrastructure [1, 4].

A primary feature of the future Internet would be to assign substrate network resources to the requested virtual networks. Therefore, virtual network embedding (VNE) is the key area in network virtualization. In order to embed/assign/map a virtual network onto the substrate/physical network, each virtual node is mapped to a physical node and each virtual link is embedded on a substrate path. A number of virtual networks (VNs) can be deployed on top of the physical network (or *substrate*), depending on the capability of the substrate and the demands of VNs.

Virtual network embedding (VNE) problem is NP-hard [9, 10] where several constraints need to be satisfied. In order to map a VN onto the substrate, requirements of both its vertices as well as edges should be fulfilled. In addition to this, VNs can arrive at different times, in any order and can be based on any standard network topology (e.g., star, bus, ring, or mesh). The substrate network also has a limited amount of resources. Thus, we need to embed or map a VN with resource constraints onto the substrate network (SN) which has finite resources.

In this paper, we evaluate three different vertex mapping approaches for VNE. Our first approach deals with mapping virtual vertices onto any available substrate nodes which can satisfy their demand. This method does not take into account the possibility of a node becoming bottleneck at the time of mapping a VN's vertex, is named as baseline approach (BLA), and was presented in [11]. Second approach is focused on mapping virtual vertices to the substrate nodes with maximum resources, is called as greedy node mapping (GNM), and was presented in [9]. The advantage of using GNM is that it can minimize the use of substrate resources from bottleneck nodes. Drawback of GNM can be that vertices may get mapped in such a way that more bandwidth resources may be needed to map a VN as compared to BLA. Third approach is being proposed in this paper and is named as HBNRM (hybrid BLA bottleneck node reduced mapping). Main focus of this new approach is to utilize benefits of both BLA and GNM while minimizing their disadvantages.

Main contributions of this paper are to evaluate different vertex mapping approaches and investigate their impact on VN embedding, how substrate's nodes become bottleneck and get exhausted. Resource utilization as a result of mapping vertices at different substrate locations by three vertex mapping approaches is also analyzed. In order to thoroughly investigate the impact of vertex mapping by any approach, evaluations are done by mapping sparsely and densely connected VNs on sparsely as well as densely connected substrate networks. The proposed solution starts with the hypothesis that it should be possible to avoid complete exhaustion of node resources at a lesser cost, which can improve VN embedding possibilities.

Rest of the paper is organized as follows. Section 2 defines the problem while Section 3 presents work done in the area of VNE. Section 4 describes our solution whereas, Section 5 presents simulation results. Section 6 concludes the paper.

2. Network Model and Problem Description

The proposed solution represents virtual as well as substrate networks as undirected graphs. The substrate network is represented by $S = (N, A^N, L, A^L)$, whereas the network to be mapped; that is, the VN is shown by $M = (V, D^V, E, D^E)$. Notations for describing the VN mapping problem are summarized in Table 1.

Throughout this document when a reference is made to a link or node, it means that it belongs to the substrate, while VN's link and node will be termed as edge and vertex, respectively. We consider central processing unit (CPU) as a resource for nodes and vertices, and bandwidth to be the resource for edges and links.

Figure 1 shows a substrate network whereas Figure 2 represents a VN request. Notation for describing node and link capacities is similar to the one proposed in [9].

VN will only be mapped on the substrate if requirements of each of its vertex as well as edge are satisfied. After mapping vertices onto the nodes which satisfy vertex demand, paths need to be calculated for each pair of nodes in the VN. Then link resources in the path are compared with the edge demand. At this point, if the path satisfies edge request, then VN is completely mapped. After satisfying requests of vertices and edges of a VN, a residual graph (S_{res}) is obtained which contains remaining capacities of nodes and links of the substrate [12].

In the beginning of VN embedding process, we initialize residual capacities of nodes and links with the following actual capacities:

$$\begin{aligned} R^N &= A^N, \\ R^L &= A^L. \end{aligned} \tag{1}$$

Therefore, when a node or link is mapping a vertex or an edge for the first time then, its residual capacity is equal to its original capacity ($r^n = a^n \wedge r^l = a^l$) and the vertex or edge demand (d^v or d^e) is matched with it. After initial mapping of a vertex or edge is made, the new residual capacity (r^n) of a node ($n \in N$) is obtained by subtracting vertex demand (d^v) from the node resource, whereas remaining capacity (r^l) of a link ($l \in L$) is found by deducting edge request (d^e) from the link resource:

$$r^n \longleftarrow r^n - d^v, \tag{2a}$$

$$r^l \longleftarrow r^l - d^e. \tag{2b}$$

Resources need to be returned to the substrate if, after mapping initial vertices or edges of a VN, there comes a point when requirements of a certain edge or vertex cannot be satisfied. This means that in such a scenario the initial or base graph (S_{base}) is regenerated:

$$r^n \longleftarrow r^n + d^v, \tag{3a}$$

$$r^l \longleftarrow r^l + d^e. \tag{3b}$$

Table 1: Notations of VNE problem.

S	Substrate network
N	Set of nodes belonging to the substrate network
A^N	Attribute associated with substrate nodes
L	Set of links joining two nodes ($n_i, n_j \in N$), of the substrate network
A^L	Attribute associated with substrate links
$P^S_{m,n}$	Substrate path from source node m, to destination node n
M	Virtual network
V	Set of vertices of virtual network
D^V	Demand associated with vertices
E	Set of edges connecting two vertices ($v_s, v_t \in V$), of the virtual network
D^E	Demand associated with edges

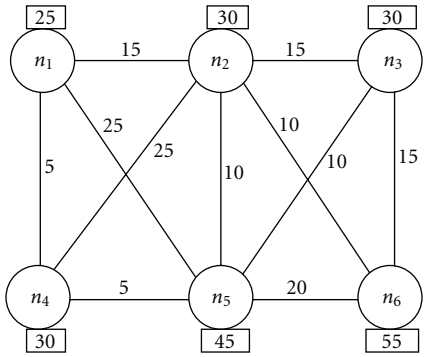


Figure 1: Substrate network.

We define the cost of mapping a VN, as sum of overall substrate resources assigned to its vertices and edges, in the same way, as previously presented in [13]. Our cost function is similar to the one given in [13]:

$$C(M) = \sum_{e \in E} \sum_{l \in L} \left(b_e^l\right) + \sum_{v \in V} (d^v). \tag{4}$$

A vertex will only be mapped on a single node whereas an edge can be mapped on a substrate path ($P^S_{m,n}$) containing one or more than one links. The term (b_e^l) in (4) indicates bandwidth allocated to an edge ($e \in E$) from a substrate link ($l \in L$).

3. Literature Review

This section is divided into two parts. It starts with a description of constraints associated with VNE, while the second part describes and categorizes work done in this area.

3.1. Constraints Associated with VNE. The virtual network embedding problem is NP-hard [9, 10] where several constraints need to be satisfied. In addition to vertex and edge constraints, the VNs can arrive at different times and in any order whereas substrate resources are also finite. Therefore, before going through the details of various solutions to the VNE problem, it is necessary to first have a look at the constraints associated with it.

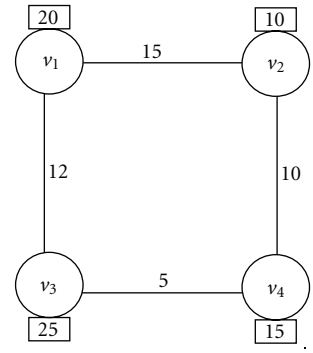


Figure 2: Virtual network.

3.1.1. Node Constraints. Two types of node constraints may be associated with a VN request.

(i) Capacity. A VN request may be constrained by a certain amount of resources on its nodes. Nodes of a VN may require a fixed number of processing or memory resources, for example, in order to run an experiment, 500 MHZ of CPU may be required for each virtual node of the VN [9].

(ii) Location. In addition to the capacity, placement of VN's nodes may be required in certain locations. This constraint may be imposed if, VN's nodes are part of a service which requires this feature, for example, CDN (Content Distribution Network), gaming service.

3.1.2. Link Constraints. Two types of link constraints may be associated with a VN request.

(i) Bandwidth. In order to run an experiment or provide a service, a VN may require certain amount of bandwidth on each of its links [9].

(ii) Link Propagation Delay. In addition to the demand of bandwidth on its links a VN may also be constrained by link propagation delay, for example, a VN carrying delay sensitive traffic to provide a service such as QoS (Quality of Service) [14].

3.1.3. Admission Control. The substrate network has finite resources on its nodes and links. Admission control process needs to be implemented for two reasons.

(i) It ensures that demands of newly arrived VNs can be fulfilled by the substrate.

(ii) Resource allocation made to already mapped VNs is not violated.

Therefore, VN requests may be rejected or postponed if the substrate does not have sufficient resources to satisfy demands of a VN at the time of arrival [9].

3.2. Virtual Network Embedding Approaches. After going through challenges associated with the VNE problem, we can now have a look at how they have been taken care of by various solutions. Details of such solutions will be given first; later on Table 2 also depicts this process. Description of each of these types of solutions is presented below.

3.2.1. Constraints. We consider node capacity, link bandwidth, and admission control (defined above) as basic constraints to the VNE problem. Some solutions to the VNE problem handle all basic constraints while others only provide solution to a subset of these constraints.

The solutions presented in [18, 20] do not perform admission control. Node capacity and admission control constraint are not considered in [16, 17]. Assuming that vertex mapping is known in advance, the authors have only provided solution for edge mapping in [19], that is, they have not taken care of node capacity constraint.

The approaches presented in [9, 10, 13, 14, 21] take care of all basic constraints.

3.2.2. Method. In order to embed a VN onto the substrate, we need to find appropriate mappings for both its vertices as well as edges. Therefore, the VNE problem can be decomposed into vertex and edge mapping, and for this, various approaches can be adopted.

(A) Vertex Mapping

(i) Iterative Method. The iterative method of mapping VNs onto substrate networks was presented in [16], where nodes are categorized as backbone and access nodes. In this method, first backbone nodes are mapped onto the substrate, then access nodes are connected to backbone nodes and shortest paths are computed between these nodes, after this, link capacities are calculated and in the end it is ensured that the backbone nodes have been mapped optimally.

The vertex mapping approach in [14] can also be considered to be iterative, as it selects one node (of highest degree) in each step. The process is repeated for remaining nodes moving on from nodes with highest to lowest degrees until all of them get mapped on the substrate.

Vertex mapping approach in [20] divides the entire VN topology into a set of elementary clusters. The decomposition of VNs is based on star topology, where nodes are characterized as hub and spoke. Mapping of a VN is done sequentially by assigning the decomposed star topology based VNs to the substrate, one at a time.

(ii) Simulated Annealing. Simulated annealing approach has been used to find the optimal topology for a given communication pattern in [17], where the goal is to find optimal reconfiguration policies.

(iii) Greedy Node Mapping. Greedy node mapping approach maps vertices on nodes with maximum resources [9, 10, 18]. The advantage of using this method is to minimize the use of substrate resources at bottleneck nodes/links, which helps in satisfying the requirements of future VN requests which demand fewer resources.

(iv) Baseline Approach (BLA). The baseline approach (BLA) of mapping vertices on any available substrate nodes which can satisfy their demand (by only evaluating if, $r^n \geq d^v$) was presented in [11]. VNs embedded using this approach can incur less cost as compared to GNM. However, BLA does not take into account the possibility of a node becoming bottleneck at the time of mapping a VN's vertex.

(v) Mixed Integer Programming. In [13], the authors have formulated a solution to the VNE problem by using mixed integer programming (MIP) formulation. Vertex mapping in this solution is done by using two techniques. In first algorithm vertex mapping is done deterministically and is called D-Vine (deterministic rounding based virtual network embedding algorithm) while second algorithm does it randomly and is presented as R-Vine (randomized rounding based virtual network embedding algorithm).

(B) Edge Mapping. The edge mapping approaches can be devised based on flows. The flows can be categorized as either unsplittable or splittable.

(i) Shortest Path Mapping (SPM). The shortest path mapping is a cost efficient approach of mapping edges on substrate paths. It has been used as a primary approach for edge mapping in a number of solutions. The solutions proposed in [14–18, 20, 21] have used SPM for edge mapping while the one given in [9] also uses it for unsplittable flows.

(ii) Multicommodity Flow. In case of splittable flows, the multicommodity flow based approach has been used for edge mapping [9, 10, 13, 19].

3.2.3. VN Requests. Virtual network requests can be either specified in advance (offline problem) or arrive as part of a dynamic process (online problem). The solutions given in [16, 18, 20] solve offline version of VNE problem, while the ones proposed in [9, 13, 14] solve it as an online problem.

3.2.4. Type of Mapping (TOM). A VN embedding algorithm may be carried out either in a distributed or centralized manner. The solutions proposed in [9, 10, 16–18, 21] map VN requests in a centralized way while the ones proposed in [15, 20] assign VN requests to substrate networks using a distributed process.

3.2.5. Adaptability. After VN requests get mapped on the substrate, a VNE solution may need to provide the feature of adaptability, that is, respond to variations in either substrates or VNs. This may be required in either of the following scenarios.

(i) A user may add new requirements for an embedded VN request. A set of new candidate resources may need to be identified in response to the additional requirements.

The above mentioned change in user's requirements was taken care of and a solution in this regard was proposed in [15]. A solution to the problem of dynamically reconfiguring topology of an overlay network in response to changes in communication requirements was also presented in [17].

(ii) A physical node/link may be hosting many virtual nodes/links. In case, a problem occurs with a single physical node/link then several virtual nodes/links will be affected. Therefore, the physical/virtual node and link failures should

Table 2: Solution to VNE constraints and approaches adopted by various proposals.

Reference	ABC		VN requests	OM	Adaptability	Method		Opt obj
	C	NC				Vertex	Edge	
[9]	✓		ONL	Cent	PM	GNM	SPM/MCF	MRU
[14]	✓		ONL	Cent	LF	IM	SPM	MQR
[15]	✓		ONL	Dist	NLF, CUR	IM	SPM	MAT
[16]		NCap, AC	OFL	Cent	NC	IM	SPM	MNC
[17]		NCap, AC	ONL	Cent	CUR	SA	SPM	MOUC
[18]		AC	OFL	Cent	CNC	GNM	SPM	MRU
[13]	✓		ONL	Cent	NC	D-Vine/R-Vine	MCF	MARR
[19]		NCap	OFL	Cent	NC	—	MCF	MAR
[20]		AC	OFL	Dist	NC	IM	SPM	MNC
[21]	✓		ONL	Cent	NC	SA	SPM	MMT
[10]	✓		ONL	Cent	Re-opt	GNM/ D-Vine/R-Vine	SPM/MCF	MAR

Where in Table 2—ABC: all basic constraints data, NC: not considered, AC: admission control, OFL: offline, Dist: distributed, LF: link failures, CUR: change in user's requirements, IM: iterative method, GNM: greedy node mapping, D-Vine: deterministic rounding based virtual network embedding algorithm, R-Vine: randomized rounding based virtual network embedding algorithm, MCF: multicommodity flow, Opt Obj: optimization objective, MQR: maximize quality and resilience, MNC: minimize network cost, MARR: maximize acceptance ratio and revenue, MAR: maximize acceptance ratio, C: considered, NCap: node capacity, ONL: online, Cent: centralized, PM: path migration, NLF: node and link failures, CNC: change in network conditions, SA: simulated annealing, SPM: shortest path mapping, TOM: type of mapping, MRU: maximize resource usage, MAT: minimize adaptation time, MOUC: minimize overlay usage cost, Re-opt: reoptimization, MMT: minimize mapping time.

always be kept into consideration and virtual nodes and links should be re-mapped if a failure occurs.

An approach to take care of vertex/node as well as edge/link failures of VNs and substrates, and remap VN's vertices and edges on alternate nodes and links has been presented in [15]. Provision of path resiliency by constructing alternate one-hop overlay routes via intermediary nodes was part of the solution proposed in [14].

(iii) The concept of "path migration" by either changing the splitting ratios of existing paths or selecting new underlying paths can enable a substrate to accommodate a newly arrived VN.

The idea of path migration was presented in [9].

(iv) After being mapped, a VN may be reconfigured to be assigned to a different set of substrate nodes and links upon arrival of a new VN request.

In [18], a solution termed as "VN assignment with reconfiguration" has been proposed which states that node and link assignments to an embedded VN request are not fixed for its lifetime and may be changed at the arrival of a new VN request in order to better utilize substrate resources.

3.2.6. Optimization Objective. VNE is a resource constrained problem and in addition to the main objective of optimizing the use of substrate resources, proposed solutions for this problem have focused on several other factors as well.

The solutions proposed in [9, 18] focus on maximizing the usage of substrate resources, while in [14] the focus has been on mapping of virtual networks to achieve high quality and resilience.

In case, a substrate node or link fails, the virtual vertices or edges mapped on it should be moved quickly enough (adaptation time should be minimized) to other nodes or links which can satisfy resource requirements, which was the objective of distributed fault-tolerant embedding algorithm, as proposed in [15].

The objective of mapping virtual networks onto a common substrate in such a way that can enable a network to support any traffic pattern allowed by a general set of constraints while minimizing the network cost was presented in [16].

Using dynamic overlay topology reconfiguration, a solution was proposed in [17] to minimize the cost of using an overlay. The two types of costs considered were occupancy cost and reconfiguration cost.

The objective of maximizing the acceptance ratio and revenue was achieved by doing coordinated node and link mapping as presented in [13].

The goal of maximizing the number of accepted VNs by preallocating resources for nodes and solving link mapping based on multicommodity flow was proposed in [19].

The objective of minimizing the network cost by mapping VNs using a distributed method and in the process achieving balanced load-sharing among all substrate nodes was the focus of solution proposed in [20].

The goal of minimizing mapping time was achieved by using simulated annealing technique and presented in [21].

Table 2 shows how solutions to the VNE problem have handled challenges associated with it.

4. Hybrid BLA-BNRM Approach (HBNRM)

VN mapping process starts by assigning vertices to nodes, then proceeds on to find k-shortest paths [22] between each pair of mapped nodes, and finishes by mapping edges onto paths that satisfy their demand. Our approach is inspired to some extent by [9] as we use similar notations to denote both virtual and substrate networks. However, we use a different k-shortest paths algorithm [22] than edge disjoint

paths [9], as it gives us a better choice of mapping an edge on substrate paths. The proposed approach solves VNE problem by considering all basic constraints (Section 3.2.1), handles VN requests online (Section 3.2.3), type of mapping is centralized (Section 3.2.4) whereas optimization objective (Section 3.2.6) is to maximize acceptance ratio (MAR). This section will initially present a description of our vertex mapping approach and in the second phase, edge mapping approach will be described.

4.1. Vertex Mapping. First step of the proposed solution starts by finding candidate nodes of the substrate, which can map vertices by satisfying their demands. In this phase, each vertex ($v \in V \wedge d^v \in D^V$) has to be mapped to a different node ($n \in N \wedge a^n \in A^N$). Several approaches can be adopted for this purpose and each will affect how VNs get mapped as well as substrate resources are utilized in the process. In this paper, two existing (i.e., BLA and GNM) and one proposed approach (HBNRM) will be evaluated.

Before going through the details of our vertex mapping approach (HBNRM), it is important to give definition of bottleneck as well as exhausted nodes.

4.1.1. Bottleneck Nodes (B(N)). The idea to minimize the use of substrate resources from bottleneck nodes and links was presented in [9], while, the concept of bottleneck links was also mentioned in [23]. Nodes and links having lack of residual capacities to map vertices/edges and hence resulting in rejection of a VN request were termed as bottlenecks in [10]. We proposed definitions for bottleneck nodes and links of a substrate in terms of their capability of mapping vertices and edges of VNs due to arrive in future in the mapping process in [11].

We define a node as a bottleneck if, it is unable to map two vertices (of highest capacity) of future VN requests. In other words, residual capacity of a bottleneck node is less than a certain value ($r^n < \text{val}_n$).

4.1.2. Exhausted Nodes (E(N)). An exhausted node is a bottleneck node, whose resources get completely utilized ($r^n = 0$).

We now describe our vertex mapping approach. In future work of [11] it was mentioned that one possible extension of that work could be to investigate how the two approaches (i.e., BLA and BNRM) could be combined in order to maximize the number of virtual networks that are mapped, while still avoiding bottleneck nodes. Another objective of the new approach should be to utilize benefits of both BLA (baseline approach) and GNM (greedy node mapping approach) while trying to minimize their disadvantages. In other words, we should be able to minimize bottleneck nodes of a substrate (GNM's advantage) while trying to minimize the cost to map a VN (BLA's advantage). We have named this approach as HBNRM (Hybrid BLA-BNRM) approach which is presented below.

One important component of HBNRM is the use of node exhaustion limit (nel) values. Nel is a value which is used to make sure that a node does not become bottleneck. The vertex is only mapped on the node if, after mapping, the node has resources equal to or greater than nel. Nel values are used according to the rule defined below.

80/50 Rule for NEL Values. We start by using a nel value (val_n, defined above) which ensures that a node does not become bottleneck after mapping a vertex. This value is increased or decreased according to the following criteria.

(i) When about eighty percent (80%) of nodes reach the set nel value in an interval (or request window), it is decreased to the next level and then same rule gets applied to the new value.

Experiments have shown that once about eighty percent nodes reach a set nel value then it may be decreased otherwise, VNs may get dropped for next interval even though sufficient node resources maybe present in the substrate.

(ii) If greater than fifty percent (50%) VNs get dropped in an interval in early stages of VN mapping process, then, nel value is increased to the next level.

When VNs get rejected in early stages of mapping process then, it could mean that sufficient node resources are present in the substrate but link resources have started to exhaust as a result of mapping vertices on same nodes repeatedly. By increasing nel value it can be ensured that nodes which were not selected previously can now be selected for mapping of new VN requests and as a result more link resources could be made available.

(iii) If greater than fifty percent (50%) VNs get dropped in an interval in later stages of VN mapping process, then, nel value is decreased to the next level.

When VNs get rejected in later stages of mapping process, then it could mean that link resources have started to exhaust and a decrease in nel value may map vertices on different nodes and as a result some unused links may become available for edge mapping.

So, nel value is increased or decreased when either of above conditions occurs. In Section 5.2, we will explain cases where nel values were increased or decreased and which condition of 80/50 rule was applied.

Another important point about 80/50 rule is that it can be modified according to number of VNs considered for an interval (request window). In this paper, 50 VNs constitute an interval (request window), if number of VNs are reduced to 40 for an interval this could change the rule to 85/55. Similarly if VN requests are increased to 60 then rule might become 75/40.

The vertex mapping function for HBNRM can be given as

$$V^M : (V, D^V) \longrightarrow (N, A^N). \quad \text{Subject to } ((r^n \geq d^v), ((r^n - d^v) \geq \text{nel}_t)). \tag{5}$$

1: Take a VN request.
2: Find a unique substrate node, for every vertex, having sufficient resources to satisfy the CPU demand,
(according to the selected vertex mapping function, BLA, HBNRM, or GNM), start from the vertex demanding most resources.
3: **If** all the vertices can be mapped at this stage, **then**, generate residual node capacities according to (2a), **else**, **GOTO** 5.
4: Call "edge mapping algorithm."
5: **If** this was the last request, **then**, **stop**, **else**, **GOTO** 1.

Algorithm 1: Vertex mapping.

1: Take the request which has successfully passed the vertex mapping stage.
2: **for** (each_edge_in_the_request) **do**
2.1: Search k-shortest paths incrementally between pair of vertices connected by the edge
(mapped on nodes by the vertex mapping algorithm).
2.2: **Stop** searching in above Step 2.1, when either edge demand is satisfied (according to edge
mapping function, SPM) or all paths have been searched.
2.3: **If**, edge request is not satisfied, **then**, **GOTO** 3, **else**, generate residual link capacities of the
selected path according to (2b) and **GOTO** 4.
2.4: **end for**
3: **If**, this was the first edge of VN, **then,** return node resources to the substrate according to
(3a), **else**, return both the link and node resources to the substrate (as defined in (3a) and (3b)).
4: **If** this was the last request, **then**, **stop**, **else,** call "vertex mapping algorithm."

Algorithm 2: Edge mapping.

The nel_t in (5), defines the time at which a VN arrives and is compared with a certain nel value. Vertex mapping function in (5) starts by checking all the vertices (V) and first selects the one which demands most processor resources ($\max(D^V)$). The benefit of doing this is that if, the substrate cannot satisfy the demand of this vertex then the mapping process stops here for this VN and requirements of remaining vertices do not need to be checked, which saves amount of computations according to number of vertices in a VN. In this way, if demand of the first vertex is satisfied then the process is repeated for all remaining vertices moving on from vertex demanding most to the least resources.

The vertex mapping algorithm is defined in Algorithm 1.

VN requests are satisfied by using first come first served (FCFS) approach and the process begins by assigning vertices to unique substrate nodes according to the selected vertex mapping function (BLA, HBNRM, or GNM). In the next step, residual capacities of nodes (selected for mapping vertices of the VN) are generated. The edge mapping algorithm is called only if, all vertex requests of a VN can be satisfied at this stage.

4.2. Edge Mapping. The next step is to map an edge ($e \in E \wedge d^e \in D^E$) on a substrate path ($P^S_{m,n}$) containing one or more than one links. In the proposed solution, k-shortest paths [22] are found for each edge. The next step is to calculate resources on a path. To achieve this objective, we take link with minimum resources in the path ($\min(P^S_{m,n})$) and match edge demand with it. If this link can satisfy edge request, then remaining links will surely be able to do that.

The approach of mapping an edge on shortest of k-shortest paths, which can satisfy its demand, termed as shortest path mapping (SPM), was presented in [9, 11]. Edge mapping function for the SPM can be given as

$$\begin{aligned} &E^M : (E, D^E) \longrightarrow P^S_{m,n}. \\ \text{Subject to} \quad &\left(d^e \leq \left(\min\left(P^S_{m,n}\right)\right)\right). \end{aligned} \tag{6}$$

The edge mapping algorithm is defined in Algorithm 2.

Edge mapping phase of the solution assigns all edges to the substrate paths and is executed number of times the total edges in a VN. It starts by finding out shortest of k-shortest paths (for $k = 1$) for each edge of the VN. In the next step, resources on that path are calculated and edge demand is matched with it. If this path has sufficient resources to satisfy the edge demand, then it is selected for mapping that particular edge. Otherwise, the process continues till either a path among the k-shortest paths can satisfy edge request or no path has sufficient resources to satisfy edge demand. In case, after mapping the initial edges of a VN, there comes a point when the requirements of a certain edge cannot be satisfied by the substrate, then the resources reserved initially for edges and vertices of a VN need to be returned to the substrate (Step 3). At this point, if there are sufficient path resources to satisfy request of all the edges then the VN is completely mapped (VN request satisfied).

5. Experimental Setup and Evaluation

This section is divided into two parts; first describes experimental setup while second presents evaluation results. The proposed solution has been implemented using Matlab.

5.1. Experimental Setup. Substrate networks have been generated using the BRITE tool [24] whereas, virtual networks have been created using Matlab.

5.1.1. Substrate Networks. The proposed solution has been tested on four different substrate networks (S_1, S_2, S_3, and S_4). Two of these networks (S_1 and S_2) consist of 100 nodes and about 500 links [9, 11, 12], while S_3 and S_4 comprise of 100 nodes and about 300 links [25]. The node resource (CPU) as well as link resource (bandwidth) is assigned different values from 10 to 100 units. The size of substrates (S_1 and S_2) can be compared with that of a medium sized ISP [9].

S_1: Node and link resources are randomly chosen from 20 to 100.

S_2: Node and link resources are randomly chosen from 10 to 100.

S_3: Node and link resources are randomly chosen from 20 to 100.

S_4: Node and link resources are randomly chosen from 30 to 100.

The substrates S_1 and S_2 have more links (are densely connected) but contain different number of node and link resources. The substrates S_3 and S_4 have fewer links (are sparsely connected) and also contain different number of node and link resources.

5.1.2. Virtual Networks. Two different sets of VNs have been mapped onto substrate networks; each set differs from the other, by number of edges.

Set 1. Number of vertices of a VN is randomly chosen between 2 and 10 and vertices are randomly connected with the probability 0.3.

Set 2. Number of vertices of a VN is randomly chosen between 2 and 10 and vertices are randomly connected with the probability 0.5.

Set 2 is similar to the setup presented in [9, 11, 12, 18] while Set 1 resembles with the one given in [11, 25]. Vertex and edge resources in both sets are randomly chosen from 1 to 5. Two sets of VNs put different demand on substrate's paths and can impact on how VNs are mapped.

5.2. Evaluation. The main focus of evaluation is to analyze the effect of using three different vertex mapping approaches (BLA, GNM, and HBNRM). Evaluation is done by mapping sparsely and densely connected VNs on sparsely and densely connected substrates. Results are presented in the form of graphs and tables which show the impact of using each approach on mapping VN requests, the way resources are utilized, nodes become bottleneck and get exhausted.

Results for densely connected substrates (S_1 and S_2) will be presented first which will be followed by a discussion about sparsely connected substrates (S_3 and S_4). The section ends with a summary of results.

5.2.1. Densely Connected Substrates (S_1 and S_2)

(a) VN Mapping ($Map(M_n)$). Figures 3 to 8 represent evaluation results, where requested VNs are shown on horizontal axis, whereas mapped VNs by evaluated approaches and cost incurred in the process are presented on vertical axis.

When a substrate is densely connected, has higher number of resources (i.e., S_1), and a set of sparsely connected VNs (VN-Set 1) is mapped on it, then mapping results are almost similar for all methods (Figure 3). VNs put less demand on substrate's paths and even if vertices are mapped on different locations in the substrate by each approach, sufficient resources are available for edge mapping and therefore, mapping results are almost similar. However, among three approaches, GNM has highest cost for VN mapping, whereas BLA has least cost (Figure 3).

An almost similar VN mapping trend (like that of Figure 3) can be seen when a set of densely connected VNs (VN-Set 2) is mapped on S_1 (Figure 4).

When the substrate is densely connected but has comparatively less number of resources (i.e., S_2) and a set of densely connected VNs (VN-Set 2) is mapped on it, then mapping results are quite different than previous substrate (S_1), as shown by Figure 5. In case of GNM, vertices can be mapped at a distance from each other and since the substrate has fewer resources so, the edge demand can become difficult to fulfill. In this case, BLA and HBNRM give almost similar mapping of VN requests (Figure 5).

Tables 3 to 20 present evaluation results for the rest of evaluation criteria where number of mapped VNs in each interval, cost incurred in the process, percentage of overall utilized resources as well as bottleneck and exhausted nodes are shown. Mapped VNs ($\text{Map}(M_n)$) and cost of mapped VNs ($C(M_n)$) are shown for each interval in Tables 3 to 20, as compared to overall mapped VNs and mapping cost, shown in Figures 3 to 8. Percentage of utilized resources ($U(S_r)$), bottleneck and exhausted nodes ($B(N)$, $E(N)$) are also important to analyze, as they represent how each approach utilizes substrate's resources.

(b) Resource Utilization ($U(S_r)$). Resource utilization for Tables 3 to 20 should be viewed based on the following factors.

(i) When same number of VNs are mapped by evaluated methods at a particular instant of time.

The actual cost comparison between any compared approaches can be seen when same number of mapped VNs are analyzed from initial set of VNs. Cost is only incurred when VNs are mapped and once VN mapping decreases so will incurred cost. Secondly, since we have used random VNs

Table 3: BLA resource utilization (VN-Set1; substrate N/W: S_1).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	1662	4.30%	9%	9%
51–100	100%	1773	4.59%	11%	11%
101–150	100%	1878	4.86%	15%	13%
151–200	100%	2056	5.32%	11%	12%
201–250	100%	2003	5.18%	12%	12%
251–300	98%	2110	5.46%	13%	14%
301–350	98%	2241	5.80%	12%	10%
351–400	94%	1932	5.00%	12%	13%
1–400	98.75%	15655	40.50%	95%	94%

Table 4: GNM resource utilization (VN-Set 1; substrate N/W: S_1).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	1956	5.06%	0	0
51–100	100%	2024	5.24%	0	0
101–150	100%	1996	5.16%	0	0
151–200	100%	2083	5.39%	0	0
201–250	100%	2071	5.36%	0	0
251–300	100%	2011	5.20%	0	0
301–350	100%	2084	5.39%	78%	0
351–400	86%	1780	4.60%	22%	9%
1–400	98.25%	16005	41.40%	100%	9%

Table 5: HBNRM resource utilization (VN-Set 1; substrate N/W: S_1).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	1703	4.41%	0	0
51–100	100%	1843	4.77%	0	0
101–150	100%	1996	5.16%	0	0
151–200	100%	1970	5.10%	0	0
201–250	100%	2165	5.60%	0	0
251–300	100%	2288	5.92%	0	0
301–350	100%	2027	5.24%	92%	0
351–400	92%	1747	4.52%	4%	0
1–400	99%	15739	40.72%	96%	0

Table 6: BLA resource utilization (VN-Set 2; substrate N/W: S_1).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2707	7%	9%	9%
51–100	100%	2998	7.76%	13%	12%
101–150	100%	3238	8.38%	14%	14%
151–200	100%	3502	9.06%	13%	13%
201–250	98%	3655	9.46%	14%	14%
251–300	68%	1627	4.21%	8%	8%
301–350	70%	1695	4.38%	8%	8%
351–400	70%	2506	6.48%	8%	8%
1–400	88.25%	21928	56.73%	87%	86%

Table 7: GNM resource utilization (VN-Set 2; substrate N/W: S_1).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	3314	8.57%	0	0
51–100	100%	3469	8.97%	0	0
101–150	98%	3338	8.64%	0	0
151–200	100%	3649	9.44%	0	0
201–250	96%	3074	7.95%	0	0
251–300	84%	2499	6.46%	0	0
301–350	74%	1943	5.03%	44%	0
351–400	60%	1213	3.14%	56%	0
1–400	89%	22499	58.20%	100%	0

Table 8: HBNRM resource utilization (VN-Set 2; substrate N/W: S_1).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2708	7.01%	0	0
51–100	100%	3052	7.90%	0	0
101–150	100%	3536	9.15%	0	0
151–200	98%	3565	9.22%	0	0
201–250	74%	2250	5.82%	0	0
251–300	80%	3035	7.85%	0	0
301–350	84%	2570	6.65%	87%	0
351–400	78%	2340	6.05%	5%	0
1–400	89.25%	23056	59.65%	92%	0

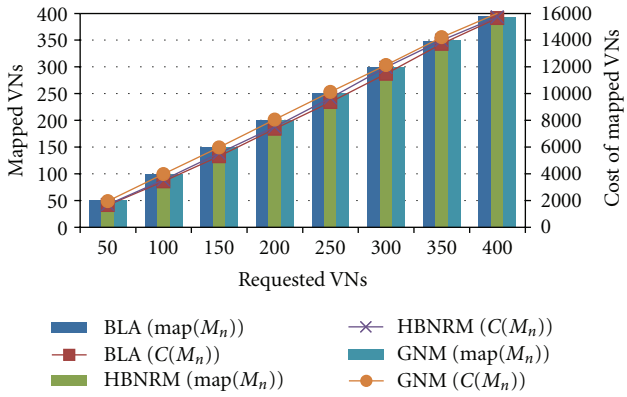


Figure 3: VN-Set 1; Substrate N/W: S_1.

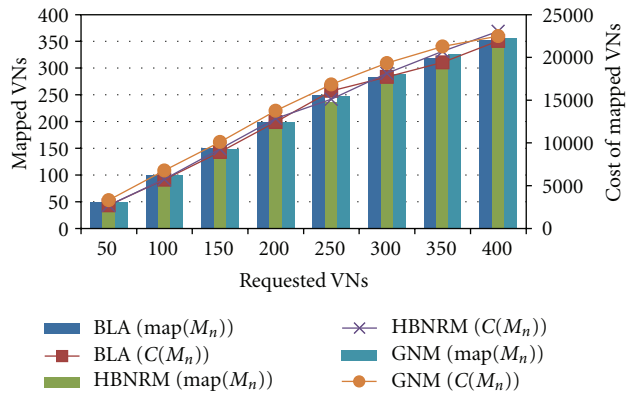


Figure 4: VN-Set 2; Substrate N/W: S_1.

so it would be unfair to compare an approach which maps more VNs having higher number of nodes and in the process incur more cost with the one which maps more VNs with lesser number of nodes and also costs less. Since, mapped VNs from initial set would be same for all the approaches so we will make cost comparison based on that.

(ii) The overall VNs mapped by using a certain method.

When more overall VNs are mapped by using a particular method then, it can incur more cost.

For sparsely connected set of VNs (VN-Set 1) three approaches map similar number of VNs when initial 250 VN requests arrive on S_1 (Tables 3 to 5). At this point, average cost of mapping a VN, $C(M_{\text{avg}})$ for BLA is 37.49, for GNM is 40.52, whereas for HBNRM is 38.71 units of substrate

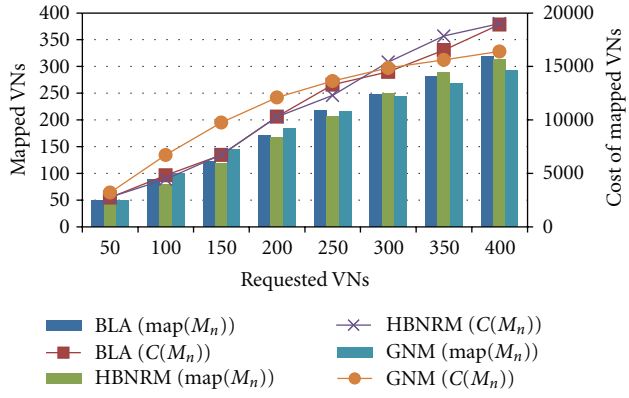


Figure 5: VN-Set 2; Substrate N/W: S_2.

Table 9: BLA resource utilization (VN-Set 2; substrate N/W: S_2).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2744	7.83%	11%	9%
51–100	76%	2080	5.93%	9%	10%
101–150	70%	1907	5.44%	10%	9%
151–200	98%	3580	10.21%	14%	15%
201–250	90%	2990	8.53%	14%	14%
251–300	60%	1198	3.42%	7%	7%
301–350	70%	2038	5.81%	11%	10%
351–400	72%	2387	6.81%	9%	9%
1–400	79.5%	18924	53.98%	85%	83%

Table 10: GNM resource utilization (VN-Set 2; substrate N/W: S_2).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	3212	9.16%	0	0
51–100	100%	3511	10.02%	0	0
101–150	90%	3035	8.66%	0	0
151–200	78%	2348	6.70%	0	0
201–250	64%	1535	4.38%	0	0
251–300	54%	1218	3.47%	0	0
301–350	52%	761	2.17%	0	0
351–400	46%	785	2.24%	0	0
1–400	73%	16405	46.80%	0	0

Table 11: HBNRM resource utilization (VN-Set 2; substrate N/W: S_2).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2775	7.92%	0	0
51–100	58%	1679	4.79%	0	0
101–150	78%	2293	6.54%	0	0
151–200	100%	3538	10.09%	0	0
201–250	78%	2010	5.73%	0	0
251–300	84%	3118	8.89%	0	0
301–350	80%	2432	6.94%	0	0
351–400	50%	1170	3.34%	76%	0
1–400	78.5%	19015	54.24%	76%	0

Table 12: BLA resource utilization (VN-Set 1; substrate N/W: S_3).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	1707	6.34%	9%	9%
51–100	100%	2181	8.09%	11%	11%
101–150	94%	2166	8.04%	13%	11%
151–200	78%	1663	6.17%	9%	11%
201–250	96%	2335	8.67%	13%	12%
251–300	94%	2257	8.38%	9%	10%
301–350	90%	2164	8.03%	14%	12%
351–400	64%	1229	4.56%	6%	7%
1–400	89.5%	15702	58.28%	84%	83%

Table 13: GNM resource utilization (VN-Set 1; substrate N/W: S_3).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2230	8.28%	0	0
51–100	100%	2274	8.44%	0	0
101–150	100%	2373	8.81%	0	0
151–200	96%	2188	8.12%	0	0
201–250	86%	2116	7.85%	0	0
251–300	88%	2110	7.83%	0	0
301–350	60%	1110	4.12%	0	0
351–400	58%	1133	4.21%	19%	0
1–400	86%	15534	57.66%	19%	0

Table 14: HBNRM resource utilization (VN-Set 1; substrate N/W: S_3).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	1753	6.51%	0	0
51–100	100%	2129	7.90%	0	0
101–150	100%	2279	8.46%	0	0
151–200	84%	1952	7.24%	0	0
201–250	100%	2476	9.19%	0	0
251–300	100%	2582	9.58%	0	0
301–350	64%	1233	4.58%	82%	0
351–400	54%	902	3.35%	1%	0
1–400	87.75%	15306	56.81%	83%	0

Table 15: BLA resource utilization (VN-Set 2; substrate N/W: S_3).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	98%	2837	10.53%	9%	9%
51–100	78%	2455	9.11%	9%	8%
101–150	28%	919	3.41%	4%	3%
151–200	42%	541	2.01%	3%	4%
201–250	42%	1080	4.01%	5%	4%
251–300	10%	183	0.68%	1%	2%
301–350	16%	651	2.42%	4%	2%
351–400	62%	1637	6.08%	5%	6%
1–400	47%	10303	38.24%	40%	38%

Table 16: GNM resource utilization (VN-Set 2; substrate N/W: S_3).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	3956	14.68%	0	0
51–100	90%	2974	11.04%	0	0
101–150	78%	2613	9.70%	0	0
151–200	52%	1019	3.78%	0	0
201–250	46%	1122	4.16%	0	0
251–300	40%	704	2.61%	0	0
301–350	24%	437	1.62%	0	0
351–400	22%	317	1.18%	0	0
1–400	56.5%	13142	48.78%	0	0

Table 17: HBNRM resource utilization (VN-Set 2; substrate N/W: S_3).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	96%	2794	10.37%	0	0
51–100	82%	2935	10.89%	0	0
101–150	52%	871	3.23%	0	0
151–200	50%	1282	4.76%	0	0
201–250	68%	1727	6.41%	0	0
251–300	60%	1808	6.71%	0	0
301–350	52%	1365	5.07%	0	0
351–400	32%	958	3.56%	0	0
1–400	61.5%	13740	51.00%	0	0

Table 18: BLA resource utilization (VN-Set 2; substrate N/W: S_4).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2884	9.92%	11%	11%
51–100	100%	3423	11.78%	13%	12%
101–150	70%	1961	6.75%	7%	7%
151–200	48%	1442	4.96%	5%	6%
201–250	60%	1821	6.27%	7%	7%
251–300	22%	304	1.05%	1%	1%
301–350	22%	582	2.00%	3%	2%
351–400	34%	1009	3.47%	3%	4%
1–400	57%	13426	46.20%	50%	50%

Table 19: GNM resource utilization (VN-Set 2; substrate N/W: S_4).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	4143	14.26%	0	0
51–100	92%	3287	11.31%	0	0
101–150	66%	2341	8.05%	0	0
151–200	64%	1916	6.59%	0	0
201–250	52%	1202	4.14%	0	0
251–300	40%	1552	5.34%	0	0
301–350	46%	1338	4.60%	0	0
351–400	22%	372	1.28%	0	0
1–400	60.25%	16151	55.57%	0	0

Table 20: HBNRM resource utilization (VN-Set 2; substrate N/W: S_4).

Req (M_n)	Map (M_n)	C (M_n)	U (S_r)	B (N)	E (N)
1–50	100%	2997	10.31%	0	0
51–100	96%	3449	11.87%	0	0
101–150	68%	2170	7.47%	0	0
151–200	48%	1688	5.81%	0	0
201–250	62%	2365	8.14%	0	0
251–300	72%	2406	8.28%	0	0
301–350	50%	1169	4.02%	0	0
351–400	48%	778	2.68%	0	0
1–400	68%	17022	58.57%	0	0

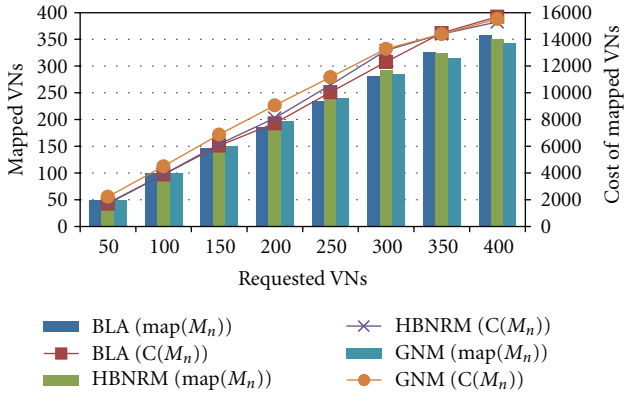


Figure 6: VN-Set 1; Substrate N/W: S_3.

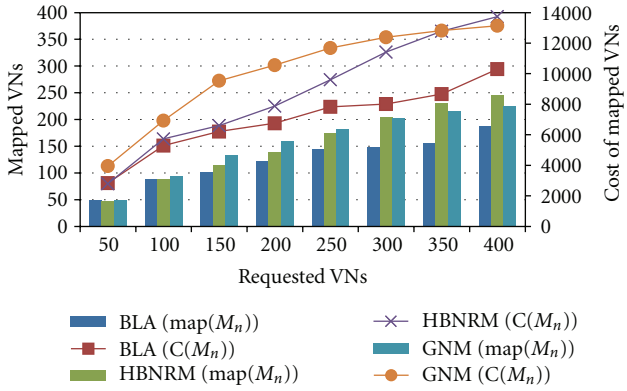


Figure 7: VN-Set 2; Substrate N/W: S_3.

resources. So BLA's cost of mapping VNs is the least whereas GNM's mapping cost is the highest among three approaches.

Overall as well, GNM uses 0.9% more resources as compared to BLA and 0.68% more when matched with HBNRM (Tables 3 to 5). For this set of VNs almost similar number of overall VNs is mapped by using either of three approaches (Tables 3 to 5).

For densely connected set of VNs (VN-Set 2) three approaches map similar number of VNs when initial 100 VN requests arrive on S_1 (Tables 6 to 8). At this point, average cost of mapping a VN, $C(M_{\text{avg}})$ for BLA is 57.05, for GNM is 67.83, whereas for HBNRM is 57.6 units of substrate resources. So, in this case as well BLA's cost of mapping VNs is least whereas GNM's mapping cost is the highest among three approaches.

Overall, HBNRM uses 2.92% more resources as compared to BLA and 1.45% more when matched with GNM (Tables 6 to 8). For this set of VNs although overall mapped VNs is almost similar by using either of three approaches but their mapping trends are different and therefore, more overall resources are utilized by HBNRM (Tables 6 to 8).

When a set of densely connected VNs (VN-Set 2) is mapped on S_2 the evaluated approaches map similar number of VNs when initial 50 VN requests arrive (Tables 9 to 11). At this point, average cost of mapping a VN, $C(M_{\text{avg}})$ for BLA is 54.88, for GNM is 64.24, whereas for HBNRM is 55.5 units

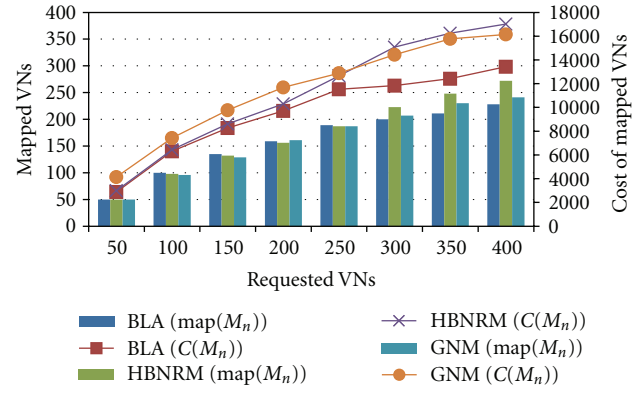


Figure 8: VN-Set 2; Substrate N/W: S_4.

of substrate resources. So, in this case as well BLA's cost of mapping VNs is the least whereas GNM's mapping cost is the highest among three approaches.

Overall, BLA uses 7.18% more resources as compared to GNM (Tables 9 and 10). However, BLA maps 6.5% more VNs as compared to GNM as well (Tables 9 and 10). When matched with HBNRM although BLA maps 1% more VNs (Tables 9 and 11) but their mapping trends are different and it uses 0.26% less resources (Tables 9 and 11).

(c) Bottleneck Nodes ($B(N)$). Nodes start to become bottleneck from VN interval-1 (VNs 1–50) when BLA is used for mapping on S_1 or S_2 (Tables 3, 6, and 9). In case of GNM there are no bottleneck nodes till the sixth interval on S_1 (Tables 4 and 7). When approach used is HBNRM then also nodes start to become bottleneck after sixth interval on S_1 (Tables 5 and 8). For HBNRM nodes do not become bottleneck as long as nel value is set according to bottleneck limit (val_n) as defined in Section 4.1. Starting nel value for nodes in Tables 5, 8, 11, 14, 17, and 20 is 10 ($\text{val}_n = 10$), since, maximum capacity of any vertex for both sets of VNs is 5. If nel value needs to be decreased to comply with either condition (i) or (iii) of 80/50 rule for nel values (Section 4.1) it is initially set to 5 for these sets of VNs ($\text{val}_n = 5$). The objective behind this new value is that a node will still be able to map at least one vertex of highest capacity of future requests. On substrate S_1 for both sets of VNs nel value needs to be decreased for seventh interval to comply with condition (i) of 80/50 rule for nel values (Tables 5 and 8). So, bottleneck nodes start to appear from there onwards.

In case of GNM there are no bottleneck nodes on S_2 (Table 10), as it maps fewer VNs than compared approaches (Tables 9 to 11). When approach used is HBNRM then, nodes start to become bottleneck after seventh interval (Table 11). According to condition (i) of 80/50 rule for nel values (Section 4.1) nel value needs to be decreased for eighth interval (Table 11). So, bottleneck nodes start to appear in that interval.

(d) Exhausted Nodes ($E(N)$). When BLA is used on S_1 or S_2, nodes start to exhaust from first interval (Tables 3, 6, and 9). In case of GNM only 9% nodes exhaust for VN-Set 1 on S_1

(Table 4) whereas no node resource exhausts for VN-Set 2 (Table 7), similar trend can be seen for HBNRM where no node resource exhausts (Tables 5 and 8). When GNM and HBNRM are used on S_2, no node resources exhaust (Tables 10 and 11).

5.2.2. Sparsely Connected Substrates (S_3 and S_4)

(a) VN Mapping ($Map(M_n)$). When a substrate is sparsely connected (i.e., S_3) and a set of sparsely connected VNs (VN-Set 1) is mapped on it, then mapping results are almost similar for all methods (Figure 6). VNs put less demand on substrate's paths and even if vertices are mapped on different locations in the substrate by each approach, sufficient resources are available for edge mapping and therefore, mapping results are almost similar (Figure 6).

When a set of densely connected VNs (VN-Set 2) is mapped on the same substrate (S_3), then mapping results are quite different than previous set of VNs (Figure 7). In case of BLA, vertices can be mapped repeatedly on same nodes and since the substrate has fewer link resources they can exhaust early. Therefore, the edge demand can become difficult to fulfill. In this case, GNM and HBNRM give almost similar mapping of VN requests until the arrival of 300 VNs (Figure 7). However HBNRM overall, maps more VNs as compared to GNM (Figure 7).

When the substrate is sparsely connected, has comparatively higher number of resources (i.e., S_4) and a set of densely connected VNs (VN-Set 2) is mapped on it, then too trend of mapping VNs by evaluated approaches is quite similar to that of previous substrate (S_3, Figure 7), as shown by Figure 8. In this case, GNM and HBNRM give almost similar mapping of VN requests until the arrival of 250 VNs (Figure 8). However HBNRM overall, maps more VNs as compared to GNM (Figure 8).

(b) Resource Utilization ($U(S_r)$). For sparsely connected set of VNs (VN-Set 1) three approaches map similar number of VNs when initial 100 VN requests arrive on S_3 (Tables 12 to 14). At this point, average cost of mapping a VN, $C(M_{\text{avg}})$ for BLA is 38.88, for GNM is 45.04, whereas for HBNRM is 38.82 units of substrate resources. In this case, BLA and HBNRM's cost of mapping VNs is almost similar whereas GNM's mapping cost is the highest among three approaches. Overall BLA uses 0.62% more resources as compared to GNM and 1.47% when matched with HBNRM (Tables 12 to 14). However, BLA maps 3.5% more VNs as compared to GNM and 1.75% more when matched with HBNRM.

When a set of densely connected VNs (VN-Set 2) is mapped on S_3 then, the evaluated approaches do not give similar mapping results from initial set of VNs (Tables 15 to 17). Therefore, cost comparison for this set of VNs cannot be presented. Overall, BLA uses 10.54% less resources as compared to GNM, and 12.76% less when matched with HBNRM (Tables 15 to 17). However in this case, BLA maps 9.5% less VNs as compared to GNM (Tables 15 and 16) and 14.5% less when matched with HBNRM (Tables 15 and 17).

Three approaches map similar number of VNs when initial 50 VN requests arrive on S_4 (Tables 18 to 20). At this point, average cost of mapping a VN, $C(M_{\text{avg}})$ for BLA is 57.68, for GNM is 82.86, whereas for HBNRM is 59.94 units of substrate resources. So, in this case as well BLA's cost of mapping VNs is least whereas GNM's mapping cost is highest among three approaches.

Overall HBNRM uses 12.37% more resources as compared to BLA (Tables 18 and 20) and 3% more when matched with GNM (Tables 19 and 20). However in this case, HBNRM maps 11% more VNs as compared to BLA (Tables 18 and 20) and 7.75% more when matched with GNM (Tables 19 and 20).

(c) Bottleneck Nodes ($B(N)$). Nodes start to become bottleneck from VN interval-1 (VNs 1–50) when BLA is used for mapping on S_3 or S_4 (Tables 12, 15, and 18). In case of GNM there are no bottleneck nodes till the seventh interval for VN-Set 1 on S_3 (Table 13), and for VN-Set 2 no node becomes bottleneck (Table 16). When approach used is HBNRM then nodes start to become bottleneck after sixth interval in case of VN-Set 1 (Table 14). For HBNRM, nodes do not become bottleneck as long as nel value is set according to bottleneck limit (val_n) as defined in Section 4.1. However, according to condition (i) of 80/50 rule for nel values (Section 4.1) it needs to be decreased for seventh interval for VN-Set 1 (Table 14). In case of VN-Set 2 no node becomes bottleneck (Table 17). Less number of VNs gets mapped for VN-Set 2 as compared to VN-Set 1 for both GNM and HBNRM and therefore no nodes become bottleneck (Tables 16 and 17). According to condition (iii) of 80/50 rule for nel values (Section 4.1) it needs to be decreased for next interval for VN-Set 2 (Table 17) if more VN requests arrive as more than 50% VNs get dropped in eighth interval (Table 17).

In case of GNM there are no bottleneck nodes on S_4 (Table 19). When approach used is HBNRM then, also there are no bottleneck nodes (Table 20). However, according to condition (ii) of 80/50 rule for nel values (Section 4.1) nel value needs to be increased for fifth interval (Table 20). Number of mapped VNs is below 50% in the fourth interval (Table 20) and therefore nel value is increased to the next level for fifth interval and again 80/50 rule is applied on that value. Moreover, according to condition (iii) of 80/50 rule for nel values (Section 4.1) it needs to be decreased for next interval if more VN requests arrive as more than 50% VNs get dropped in eighth interval (Table 20).

(d) Exhausted Nodes ($E(N)$). When BLA is used, nodes start to exhaust from first interval on S_3 and S_4 (Tables 12, 15, and 18). In case of GNM no node resource exhausts on S_3 and S_4 (Tables 13, 16, and 19), similar trend can be seen for HBNRM where no node resource exhausts (Tables 14, 17, and 20).

Summary 1. When a substrate has higher number of resources and is densely connected, then the three approaches give almost similar results in terms of number of mapped VNs when either sparsely or densely connected VNs are mapped (Figures 3 and 4, Tables 3 to 8). Resources

by each approach are utilized in a different manner. BLA uses least whereas GNM uses highest number of resources (Figures 3 and 4, Tables 3 to 8). However, when the substrate is densely connected but has lesser number of resources and a set of densely connected VNs are mapped then BLA maps more VNs as compared to GNM (Figure 5, Tables 9 and 10).

On a sparsely connected substrate, when a set of sparsely connected VNs get mapped then also compared approaches give almost similar mapping results (Figure 6, Tables 12 to 14). However, when the substrate is sparsely connected but a set of densely connected VNs are mapped then GNM maps more VNs as compared to BLA (Figures 7 and 8, Tables 15 and 16, Tables 18 and 19).

The HBNRM approach is either close to or gives better VN mappings than compared approaches on either sparsely or densely connected substrates (Figures 3 to 8, Tables 3 to 20). The flexibility of 80/50 rule for nel values facilitates in doing better vertex mapping in changing scenarios and thus good mapping results can be achieved regardless of the type of substrate. HBNRM comes close to GNM in terms of minimizing complete exhaustion of node resources of a substrate, and also is near to BLA in terms of reducing mapping cost of VNs (Tables 3 to 20).

6. Conclusion and Future Work

We have proposed an approach to virtual network embedding which not only minimizes complete exhaustion of substrate nodes but also does that at the cost of utilizing comparatively less resources than an existing approach. Main focus of this approach is to do cost efficient mapping of vertices on those nodes of a substrate which after mapping, do not become bottleneck for future VN requests.

The proposed approach (referred to as HBNRM) has been compared with existing vertex mapping methods BLA and GNM. BLA does not take node resource exhaustion into consideration which GNM does but can map VNs at a higher cost. The number of virtual networks that can be assigned to a substrate has been investigated for varying distributions of VN requests and substrate topologies. The results show that BLA is favorable for densely connected substrates, while GNM gives better results for sparsely connected ones. HBNRM, on the other hand either gives almost similar or better VN mappings for both sparsely as well as densely connected substrates when compared with BLA and GNM.

One possible extension of this work is to include the feature of adaptability to either deal with change in user's demands after a VN gets mapped on the substrate or, handle node and link failures.

References

[1] J. S. Turner and D. E. Taylor, "Diversifying the Internet," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'05)*, pp. 755–760, December 2005.

[2] F. A. Shaikh, S. McClellan, M. Singh, and S. K. Chakravarthy, "End-to-end testing of IP QoS mechanisms," *Computer*, vol. 35, no. 5, pp. 80–87, 2002.

[3] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the internet impasse through virtualization," *Computer*, vol. 38, no. 4, pp. 34–41, 2005.

[4] N. Feamster, L. Gao, and J. Rexford, "How to lease the Internet in your spare time," *SIGCOMM Computer Communication Reiew*, vol. 37, no. 1, pp. 61–64, 2007.

[5] A. Nakao, "Network virtualization as foundation for enabling new network architectures and applications," *IEICE Transactions on Communications*, vol. E93-B, no. 3, pp. 454–457, 2010.

[6] A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford, "In VINI Veritas: realistic and controlled network experimentation," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '06)*, pp. 3–14, Pisa, Italy, 2006.

[7] http://www.geni.net/.

[8] Planetlab, http://www.planet-lab.org/.

[9] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding: substrate support for path splitting and migration," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 17–29, 2008.

[10] N. Farooq Butt, M. Chowdhury, and R. Boutaba, "Topology-awareness and reoptimization mechanism for virtual network embedding," *Lecture Notes in Computer Science*, vol. 6091, pp. 27–39, 2010.

[11] A. Razzaq, P. Sjödin, and M. Hidell, "Minimizing Bottleneck Nodes of a Substrate in Virtual Network Embedding," in *In Proceedings of the 2nd IFIP International Conference Network of the Future (NoF'11)*, Paris, France, November 2011.

[12] J. Lischka and H. Karl, "A virtual network mapping algorithm based on subgraph isomorphism detection," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '09)*, 2009.

[13] N. M. Mosharaf, K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual network embedding with coordinated node and link mapping," in *28th Conference on Computer Communications (INFOCOM '09)*, pp. 783–791, April 2009.

[14] J. Shamsi and M. Brockmeyer, "QoSMap: QoS aware mapping of virtual networks for resiliency and efficiency," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'07)*, November 2007.

[15] I. Houidi, W. Louati, D. Zeghlache, P. Papadimitriou, and L. Mathy, "Adaptive virtual network provisioning," in *Proceedings of the 2nd ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures*, pp. 41–48, ind, September 2010.

[16] J. Lu and J. Turner, "Efficient mapping of virtual networks onto a shared substrate," Tech. Rep. WUCSE-2006-35, Washington University, 2006.

[17] J. Fan and M. H. Ammar, "Dynamic topology configuration in service overlay networks: a study of reconfiguration policies," in *25th IEEE International Conference on Computer Communications (INFOCOM '06)*, April 2006.

[18] Y. Zhu and M. Ammar, "Algorithms for assigning substrate network resources to virtual network components," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, April 2006.

[19] W. Szeto, Y. Iraqi, and R. Boutaba, "A multi-commodity flow based approach to virtual network resource allocation," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'03)*, pp. 3004–3008, December 2003.

[20] I. Houidi, W. Louati, and D. Zeghlache, "A distributed virtual network mapping algorithm," in *IEEE International Conference on Communications (ICC '08)*, pp. 5634–5640, May 2008.

[21] R. Ricci, C. Alfeld, and J. Lepreau, "A solver for the network testbed mapping problem," *ACM Computer Communication Review*, vol. 33, no. 2, pp. 65–81, 2003.

[22] J. Y. Yen, "Finding the K shortest loopless paths in a network," *Management Science*, vol. 17, no. 11, pp. 712–716, 1971.

[23] Y. Zhu, *Routing, resource allocation and network design for overlay networks [Ph.D. thesis]*, College of computing, Georgia Institute of Technology, 2006.

[24] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: an approach to universal topology generation," in *Proceedings of the 9th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '01)*, pp. 346–353, August 2001.

[25] A. Razzaq and M. S. Rathore, "An approach towards resource efficient virtual network embedding," in *Proceedings of the 2nd International Conference on Evolving Internet (Internet '10)*, pp. 68–73, esp, September 2010.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 232797, 12 pages
doi:10.1155/2012/232797

Research Article

Bit Rate Optimization with MMSE Detector for Multicast LP-OFDM Systems

Ali Maiga,[1] Jean-Yves Baudais,[2] and Jean-François Hélard[1]

[1] *INSA, IETR, Université Européenne de Bretagne, UMR 6164, 35708 Rennes, France*
[2] *The National Center for Scientific Research (CNRS), Institute of Electronics and Telecommunications of Rennes (IETR), UMR 6164, 35708 Rennes, France*

Correspondence should be addressed to Jean-Yves Baudais, jean-yves.baudais@insa-rennes.fr

Received 14 February 2012; Accepted 21 April 2012

Academic Editor: Yi Su

We propose a new resource allocation algorithm with minimum mean square error (MMSE) detector for multicast linear precoded orthogonal frequency division multiplexing (LP-OFDM) systems. To increase the total multicast bit rate, this algorithm jointly uses the LP-OFDM modulation technique and an adaptation of the OFDM-based multicast approaches to exploit the transmission link diversities of users. The LP technique applied to multicast OFDM systems with zero forcing (ZF) detector has already proved its ability to increase the unirate multicast system bit rate in a power line communication (PLC) context. The new MMSE detector and the new related bit-loading algorithm are developed to enhance the ZF detector results. To improve both the bit rate and the fairness among multicast users, the utilization of the LP component in multirate multicast systems is then investigated. Simulations are run over indoor PLC channels, and it is shown that the proposed LP-based methods outperform the OFDM-based methods in terms of total bit rate and fairness index for both unirate and multirate multicast systems. Additionally, it is shown that the proposed bit-loading algorithm with MMSE detector outperforms the ZF detector and the OFDM-based receiver in terms of total multicast bit rate and fairness among users.

1. Introduction

Multicasting is a network addressing method for the delivery of data to a group of users simultaneously. This technique offers a significant improvement compared to unicasting because it uses less network resources. Multicast routing is a well-investigated subject in the literature for both wired and wireless systems [1]. In multiuser communication systems, all users share the same downlink resources [2]. The allocation algorithms in multicast must adapt to the system parameters to satisfy all requirements of users. In this paper, the adaptation of the physical (PHY) layer parameters is addressed for multicast orthogonal frequency division multiplexing (OFDM) systems in indoor powerline communications (PLC) context. Since indoor PLC is being used to deliver triple play services, the multicast may be interesting in this case. Over the PHY layer, resources have to be allocated in order to satisfy requirements of each multicast user. However, the difference in link conditions of users makes it difficult to adapt the PHY layer (coding rate, modulation index, etc.) to the link conditions of each user. The conventional resource allocation method in multicast OFDM consists in adjusting the PHY parameters to serve the user who experiences the worst channel condition. Consequently, all users receive the same bit rate, and this final multicast bit rate is limited by the worst user channel conditions [3].

To increase the total multicast bit rate and to better fit the link conditions, the concept of heterogeneous multicast (also called multirate multicast) was brought in [4]. The conventional multicast system refers to unirate multicast system. In multirate multicast, users are grouped into subgroups, and the receivers of a multicast subgroup are offered services at different rates commensurate with their capabilities. Therefore, multirate schemes have a great advantage over unirate multicast in adapting to diverse receiver requirements and heterogeneous network conditions [5]. One way of attaining multirate multicast is by hierarchical encoding or layered

streaming, which is particularly suitable for audio and video traffic. In this approach, the sender provides data in several layers organized in a hierarchy. Receivers subscribe to the layers cumulatively to provide progressive refinements [3, 6, 7]. Multicast users are separated into subgroups in frequency domain [3, 7], or in time domain [8]. In frequency domain and for OFDM systems, each subcarrier is assigned to a subgroup of users and carries the same data symbols. The number of loaded bits on each subcarrier is then determined considering the lowest one among the channel amplitudes of all the users sharing this subcarrier. It has been shown that this method significantly increases the total multicast bit rate compared to the conventional method in wireless communications, but degrade the fairness among users [7]. To enforce the fairness performance while minimizing throughput degradation, a subcarrier and bit allocation scheme for proportional fairness (PF) has been also proposed [7].

In [9–11], we proposed to exploit the transmission link diversities of users by jointly using the linear precoded OFDM (LP-OFDM) modulation technique and an adaptation of the conventional resource allocation scheme. The proposed resource allocation methods were applied only for unirate and time domain multirate multicast systems. LP-OFDM is a combination of multicarrier and spread spectrum techniques also known as MC-SS techniques in wireless applications. The proposed resource allocation algorithms are developed for an LP-OFDM system where the equalization is performed according to the zero forcing (ZF) criterion. The ZF detection technique consists in reversing the channel coefficients to fully correct the phase shift and the attenuation, and to completely cancel the interference between precoding sequences, but at the cost of increasing the noise level. Using this detector, we showed that the proposed algorithms offer a significant bit rate gain compared to the conventional resource allocation method in unirate multicast context. Furthermore, for multirate multicast systems, the proposed method is the most suitable method considering both the bit rate and the fairness index when users experience similar channels [11]. It has been shown that the minimum mean square error (MMSE) detector outperforms the ZF technique [12]. The MMSE detector corrects the phase shift and the attenuation of the channel fading taking into account the present signal-to-noise ratio (SNR) [13].

In this paper, we propose a new resource allocation scheme with MMSE detection technique for multicast LP-OFDM systems. In addition to the studies done in [10, 11], the linear precoding component is also applied in frequency domain multirate multicast systems to improve both the bit rate and the fairness performances. Numerical results show that the proposed LP-based methods outperform the OFDM-based methods both in unirate and multirate multicast systems. Additionally, it is shown that the new bit-loading algorithm with MMSE detector offers the best performances in terms of total multicast bit rate and fairness among users.

This paper is organized as follows. Section 2 describes the linear precoded multicarrier systems and the achievable bit rate for LP-OFDM systems using ZF and MMSE detectors. Section 3 describes the multicast systems. Section 4 presents the resource allocation algorithms in unirate multicast systems. The study of the multirate multicast systems is done in Section 5. The performance comparisons of all algorithms are given in Section 6 over PLC channels. Finally, Section 7 concludes the paper.

2. Linearly Precoded Multicarrier Systems

2.1. System Description. Multicarrier modulation techniques like OFDM for wireless and discrete multitone (DMT) for wireline have been selected as major modulation schemes, which are able to ensure high data rates in frequency-selective channels. The utilization of adaptive multicarrier systems allows to dynamically distribute the information over the subcarriers of the signal based on the value of the SNR of each subcarrier. However, when the signal power spectral density (PSD) is limited, as in PLC systems, there is a loss of quantification due to discrete modulation orders. This loss is minimized by transmitting the information on subsets of subcarriers instead of individual subcarriers [14]. The linear precoding technique consists in connecting a subset of subcarriers with precoding sequences to mutually exploit their capabilities [15]. If judiciously done, each resulting subset holds an equivalent SNR such that the total supported throughput is greater than the sum of the individual throughputs supported by each subcarrier taken separately. In the following, the subset of subcarriers is known as block and the subcarriers in one block are not necessarily adjacent. The number of blocks B is the ratio of the total number of subcarriers N to the length L of the precoding sequences. The length L is assumed to be the same for all blocks.

Figure 1 gives the LP-OFDM transmitter-receiver model. This figure shows the various operations involved when setting up an LP-OFDM signal. From the classical OFDM systems, the linear precoding matrix which is a Hadamard matrix is added. The resource allocator dynamically distributes the bits and the powers based on the channel conditions and the quality of service (QoS) requirements. In this figure, r_u and E_u are respectively the number of bits and the power allocated to the precoding sequence u. U is the number of precoding sequence, N_0 the background noise level, h_n represents the channel coefficient on subcarrier n. The PSD constraint in this context is written as

$$\sum_{u=1}^{U} E_u \leq E_{\text{PSD}}, \tag{1}$$

where E_{PSD} is the power per symbol imposed by the PSD limit.

2.2. Mutual Information of the LP-OFDM System. The frequency characteristics of the LP-OFDM signal are the same as those of the OFDM signal. The multicarrier OFDM signal is assumed to be adapted to the channel. The guard interval, the number N of subcarriers, and the carrier spacing are selected to perfectly absorb the multipaths caused by the channel

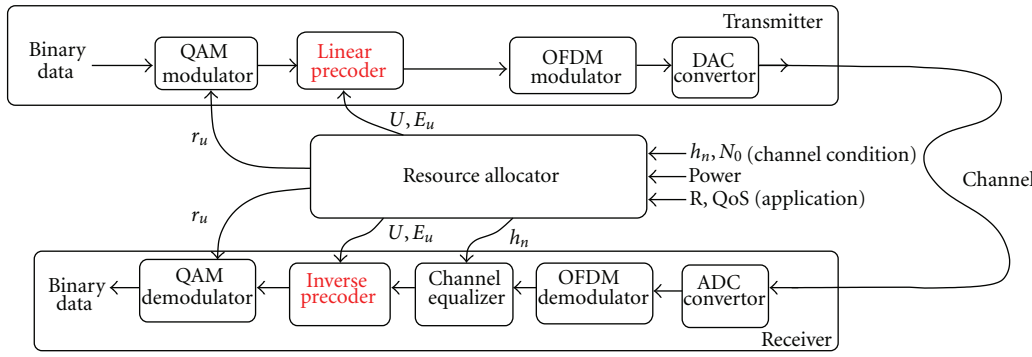


Figure 1: Transmitter-receiver model.

and to limit the loss of spectral efficiency due to the guard interval. Under these assumptions, the signal received after the OFDM demodulator can be expressed as

$$\mathbf{Y} = \mathbf{HCX} + \mathbf{Z}, \tag{2}$$

where $\mathbf{X}$ is the vector of size N of transmitted symbols before OFDM modulation and carrying the information, $\mathbf{C}$ is the precoding matrix composed of orthogonal Hadamard matrices, $\mathbf{H} = \text{diag}(h_1, \ldots, h_N)$ the diagonal matrix of the channel transfer function, $\mathbf{Z}$ is the noise vector such that $\mathbb{E}[\mathbf{ZZ}^H] = N_0\mathbf{I}_N$, and $\mathbf{Y}$ is the demodulated OFDM signal. In the general case, the estimated symbols, after the equalization and the inverse linear precoding processes, are written as

$$\hat{\mathbf{X}} = \mathbf{W}^H\mathbf{Y}, \tag{3}$$

where $\mathbf{W}$ represents the matrix of the complex equalization coefficients and the operator $(\cdot)^H$ the Hermitian conjugate. In general case, $\mathbf{W}$ can be written [16] as

$$\mathbf{W} = \mathbf{CG}^H, \tag{4}$$

where $\mathbf{G}$ is the equalization matrix.

As previously stated, we consider in this paper, the ZF and MMSE detection technique for the equalization process. Using linear precoding sequences and for full-loaded system, that is, $U = L$, the equalization matrix $\mathbf{G}$ is then a diagonal matrix [12]. The equalization coefficients g_n based on ZF and MMSE criteria are then written as [13]:

$$\begin{aligned} \text{ZF:} \quad & g_n = \frac{1}{h_n}, \\ \text{MMSE:} \quad & g_n = \frac{h_n^*}{|h_n|^2 + N_0/E_{\text{PSD}}}. \end{aligned} \tag{5}$$

The inverse precoding process is performed block per block without dependence between block. In one block of L subcarriers $l \in [1; L]$, the received symbol for the precoding sequence v, after the inverse linear precoding process, is written as:

$$\hat{x}_v = \underbrace{\sum_{l=1}^{L} c_{v,l} g_l h_l c_{l,v} x_v}_{A_1} + \underbrace{\sum_{l=1}^{L} \sum_{\substack{u=1 \\ u \neq v}}^{U} c_{v,l} g_l h_l c_{l,u} x_u}_{A_2} + \underbrace{\sum_{l=1}^{L} c_{v,l} g_l z_l}_{A_3}. \tag{6}$$

In this expression, there are, from left to right, the term A_1 of the useful signal, an interference term A_2 and a noise term A_3.

Under the assumption of simple linear receiver with independent sequence demodulation, the system capacity is expressed as the sum of the capacities provided by each precoding sequences. It is then sufficient to calculate the mutual information $\mathcal{I}_v$ between processes $\hat{x}_v$ and x_v. Notice that without independent sequence demodulation, the channel capacity is given by the maximum of the mutual information between the input and output of the channel, where the maximization is with respect to the input distribution, and the capacity is the capacity of the Gaussian interference channel [17]. In the case of ZF receiver, the assumption of independent sequence demodulation is not needed since the transmission is orthogonal, $A_2 = 0$. The mutual information $\mathcal{I}_v$ is then as follows:

$$\mathcal{I}_v = \log_2\left(1 + \frac{\mathbb{E}\left[A_1 A_1^H\right]}{\mathbb{E}\left[A_3 A_3^H\right]}\right). \tag{7}$$

Using (5), the development of the mathematical expectation terms in the mutual information leads to

$$\mathcal{I} = \sum_{v=1}^{U} \log_2\left(1 + \frac{L^2}{\sum_{l=1}^{L}\left(1/|h_l|^2\right)} \frac{E_u}{N_0}\right). \tag{8}$$

In the case of MMSE, $A_2 \neq 0$ and the mutual information under independent sequence demodulation is

$$\mathcal{I} = \sum_{v=1}^{U} \log_2\left(1 + \frac{\phi E_v}{\sum_{\substack{u=1 \\ u \neq v}}^{U} \varphi_{u,v} E_u + \lambda N_0}\right). \tag{9}$$

with

$$\begin{aligned} \phi &= \left| \sum_{l=1}^{L} \frac{|h_l|^2}{|h_l|^2 + N_0/E_{\text{PSD}}} \right|^2, \\ \varphi_{u,v} &= \left| \sum_{l=1}^{L} \frac{|h_l|^2}{|h_l|^2 + N_0/E_{\text{PSD}}} c_{v,l} c_{u,l} \right|^2, \\ \lambda &= \sum_{l=1}^{L} \frac{|h_l|^2}{\left(|h_l|^2 + N_0/E_{\text{PSD}}\right)^2}. \end{aligned} \tag{10}$$

Optimization procedures for the maximisation of the system bit rate can then be applied on these mutual information expressions.

2.3. LP-OFDM System Bit Rate. The mathematical expressions obtained with MMSE detector (9) have a form such that the studied optimization problems can be shown to be almost intractable or leads to a prohibitive complexity. The proposed bit-loading algorithm with MMSE detector uses as an input the bit distribution obtained with ZF detector. This choice will be justified later. The optimization of the LP-OFDM systems using the ZF detection technique has been studied in many works [15, 18]. Here, we provide the main results on the bit rate optimization procedures. The optimum achieved bit rate of the LP-OFDM system, under assumption of perfect synchronization, perfect channel estimation, PSD constraint and unconstrained modulations, is [15]

$$\mathcal{R} = \sum_{b=1}^{B} \mathcal{R}_b = \sum_{b=1}^{B} L \log_2\left(1 + \frac{1}{\Gamma}\frac{L}{\sum_{n\in S_b}\left(1/|h_n|^2\right)}\frac{E_{\mathrm{PSD}}}{N_0}\right), \tag{11}$$

where $|h_n|$ is the channel amplitude of subcarrier n, Γ is the SNR gap, and S_b is the subset of subcarriers within the bth block of size L. The SNR gap Γ defines the gap between a practical coding and modulation scheme and the channel capacity. This SNR gap Γ depends on the used coding and modulation scheme, and also on the target probability of error. Following the conventional SNR gap analysis [19], Γ has a constant value for all the modulation orders of uncoded quadrature amplitude modulation (QAM) and for a fixed target symbol error rate (SER). Γ is approximated by

$$\Gamma = \frac{1}{3}\left(Q^{-1}\left(\frac{\mathrm{SER}}{4}\right)\right). \tag{12}$$

To maximize the bit rate $\mathcal{R}$ given by (11), it is sufficient to minimize the sum $\sum_{n\in S_b}(1/|h_n|^2)$ for each block b. This minimization corresponds to the selection of available subcarriers with the best channel amplitudes $|h_n|$ for each block. A simple solution is then to sort subcarriers in descending order and to choose the first L subcarriers for the first block, the following L subcarriers for the second block, and so forth [15].

For real systems, the achieved bit rate using discrete modulation is maximized if, on block S_b, r_u bits are allocated to precoding sequence u, and r_u is expressed as [15]

$$r_u = \begin{cases} \left\lfloor \dfrac{\mathcal{R}_b}{L} \right\rfloor + 1 & \forall u \in [1; n_u] \\ \left\lfloor \dfrac{\mathcal{R}_b}{L} \right\rfloor & \forall u \in (n_u; L], \end{cases} \tag{13}$$

where

$$n_u = \left\lfloor L\left(2^{\mathcal{R}_b/L - \lfloor \mathcal{R}_b/L \rfloor} - 1\right)\right\rfloor. \tag{14}$$

In this paper, we propose a new bit-loading algorithm with MMSE detector for LP-OFDM systems. According to (9), to determine the optimal distribution of the power on each subcarrier, an initial power allocation is needed. Knowing that the MMSE detector outperforms the ZF technique [12] and considering the bit allocation provided in (13) as the optimal solution for ZF detector, we define this ZF result as the initialisation point. Hence, using (9), the allocated powers $\{E_u\}_{u\in[1;U]}$ to precoding sequences for a given bit distribution $\{r_u\}_{u\in[1;U]}$ in a block of subcarriers, satisfy

$$\frac{1}{\Gamma}\phi\frac{E_u}{E_{\mathrm{PSD}}} - (2^{r_u} - 1)\sum_{\substack{v=1 \\ v\neq u}}^{U} \varphi_{u,v}\frac{E_v}{E_{\mathrm{PSD}}} = (2^{r_u} - 1)\lambda\frac{N_0}{E_{\mathrm{PSD}}} \tag{15}$$

for all $u \in [1; U]$.

Let Φ, Λ, and Ψ the matrices defined by

$$\begin{aligned} &\Phi_{u,u} = \frac{\phi}{\Gamma}; \\ &\forall u \neq v, \quad \Phi_{u,v} = -(2^{r_u} - 1)\varphi_{u,v}; \\ &\Lambda_u = (2^{r_u} - 1)\lambda\frac{N_0}{E_{\mathrm{PSD}}}; \\ &\Psi_u = \frac{E_u}{E_{\mathrm{PSD}}}. \end{aligned} \tag{16}$$

Then, the distribution of relative energies $\{\Psi_u\}_{u\in[1;U]}$ satisfies

$$\Psi = \Phi^{-1}\Lambda. \tag{17}$$

In practice, the matrix Φ is diagonally dominant and then invertible. In fact, the used distribution strategy of subcarriers into blocks leads to low intersequence interference. Notice that this strategy consists in selecting subcarriers with similar amplitudes in each block in order to reduce the distortion into the blocks. As a result, the intersequence interference terms are minimized, and then Φ is diagonally dominant. The proposed algorithm consists in iteratively updating the bit distribution, while the PSD constraint is satisfied. Namely, while the sum of the relative energies $\{\Psi_u\}_{u\in[1;U]}$ is lower than one, the algorithm adds one bit more to the bit distribution $\{r_u\}_{u\in[1;U]}$, while minimizing the dispersion of values in the block. Algorithm 1 gives the new bit-loading algorithm with MMSE detector for LP-OFDM systems.

This proposed Algorithm 1 exploits one optimal condition in ZF detector: the minimization of the dispersion of the bit distribution $\{r_u\}_{u\in[1;U]}$. This condition becomes an heuristic with MMSE detector and the convergence of the algorithm to the optimal solution is not ensure. Nevertheless, the MMSE detector can outperforms the ZF detector. Furthermore, the complexity of the algorithm is reduced using the bit-rate obtained with ZF detector as initialisation process.

for all block $b \in [1; B]$ **do**
  Compute $\{r_u\}_{u\in[1;U]}$ from (11)
  Compute $\{\Psi_u\}_{u\in[1;U]}$ from (15)
  while $\sum_u \Psi_u \leq 1$ **do**
    Add 1 bit to $\{r_u\}_{u\in[1;U]}$, while minimizing the dispersion of values
    Compute $\{\Psi_u\}_{u\in[1;U]}$ from (15)
  end while
end for

Algorithm 1: Bit-loading algorithm with MMSE detector.

3. Resource Allocation in Multicarrier Multicast Systems

3.1. Systems Description. Multicast delivers data to a group of users by a single transmission, which is particularly useful for high-data-rate multimedia services due to its ability to save the network resources [3]. Figure 2 illustrates a simple multicast case where the source (in green) sends multimedia data to three receivers (in red). The source sends a known sequence of data that allows the receivers to perform the initial channel estimation. A feedback path from each receiver to the source reports the channel amplitudes $|h_{k,n}|$ on each subcarrier. We assume that the source knows perfectly the channel amplitude of all users. Based on this channel estimation and the QoS requirements, the source can perform the multicast resource allocation based on OFDM or LP-OFDM modulation technique.

Figure 3 shows the PHY-MAC cross-layer modules in the transmitter for the multicast data transmission. The channel state information of receivers is used by the multicast scheduler, the multicast subgroup management, and the resource controller. In multirate multicasting context, the multicast subgroup manager groups the multicast users into subgroups. Moreover, the multicast subgroup manager offers information such as the bit rate to the multicast scheduler module. Multicast scheduler module determines the quantity of data in every frame. The resource controller assigns time slots and subcarriers and determines the modulation order on each subcarrier. Video source encoder module encodes the streaming video data with the determined data rate and coding rate. A sufficiently large size of buffer to store the real-time data is assumed. In the message frame, the data is made from the combination of encoded video and multicast users management information [20]. Then, after processing in the PHY interface module, the multicast TX bits are transmitted to the multicast users. Notice that the guard interval is assumed to be selected to perfectly absorb the multipaths caused by all the channels. Due to the PSD constraint in PLC systems, all users have the same peak power constraint E_{PSD} on each subcarrier. Hence, there is no power allocation. To simplify calculations, it is assumed that all users utilize the same length L of precoding sequences for all blocks.

3.2. Concept of Equivalent Multicast Channel. In multicast systems, all users basically receive the same resources. The multicast bit rate could be considered as the bit rate computed for one user on an equivalent channel. This equivalent channel is derived from the different channels of users. For multicast OFDM systems, the equivalent amplitude of this channel on each subcarrier is given by the worst user subcarrier amplitude [9]. Here, we extend this concept of equivalent channel to LP-OFDM systems. The general formulation of the equivalent channel is given as

$$h_b^{\text{eq}} = f\left(\{h_{k,i}\}_{i\in\mathcal{S}_b\ k\in[1;K]}\right), \tag{18}$$

where b represents the subcarrier index in OFDM case or the block index in LP-OFMD case, and K is the number of multicast receivers. ith OFDM system, the size of $\mathcal{S}_b$ is one and $\mathcal{S}_b = \{b\}$ with $b \in [1; N]$. Computing this equivalent multicast channel, the multicast resource allocation is simplified to a single link resource allocation. Therefore, the bit-loading algorithms can be applied on this equivalent channel.

4. Unirate Multicast Systems

The unirate multicast systems refer to the multicast systems where all users receive exactly the same bit rate. To ensure such a common bit rate to all users, the resource allocations methods must adjust the PHY layer parameters to the worst user link conditions. The modulation orders for each dimension are then computed using the worst user channel gains on this dimension. Notice that the dimension corresponds to the subcarrier in OFDM case or the block of subcarriers in LP-OFDM case.

4.1. Conventional Multicast Resource Allocation. The conventional method in multicast OFDM, LCG (low channel gain [7]), consists in allocating resources while satisfying requirements of all users. This method sets the number of loaded bits per subcarrier with the lowest number of loaded bits over this subcarrier, considering all the channels of users. The equivalent channel, considered as the equivalent OFDM channel, then writes [11]

$$\left|h_n^{\text{eq}}\right|^2 = \min_u |h_{u,n}|^2. \tag{19}$$

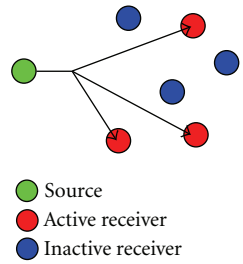


Figure 2: Multicast communication scenario.

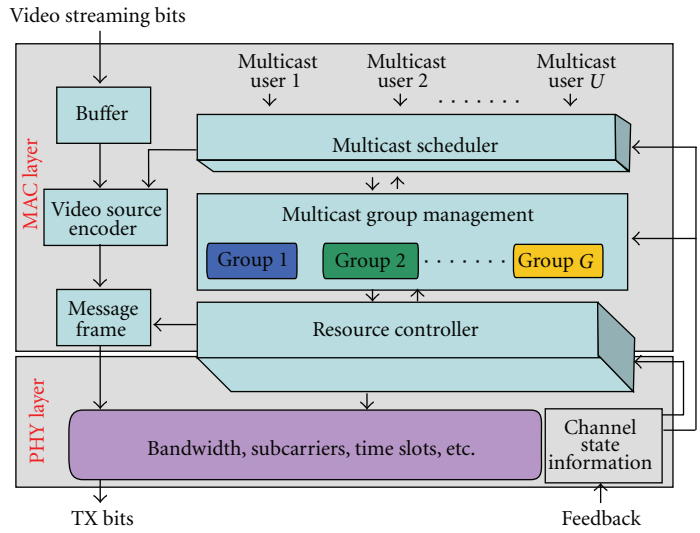


Figure 3: PHY-MAC cross-layer modules in the transmitter side [10].

Based on this channel, the multicast bit rate achieved by the LCG method, on subcarrier n, writes under PSD constraint and unconstrained modulation

$$\mathcal{R}_n^{\mathrm{LCG}} = \log_2\left(1 + \frac{E}{\Gamma N_0}\left|h_n^{\mathrm{eq}}\right|^2\right). \tag{20}$$

It has been shown that the conventional multicast bit rate is saturated as the number of users increases due to the different channel selectivity [7, 10].

4.2. Resource Allocation for Unirate Multicast LP-OFDM Systems. To increase the bit rate offered by the LCG method, we proposed new resource allocation algorithms for unirate multicast OFDM systems in indoor powerline communication context. These algorithms jointly use the LP-OFDM modulation technique with zero forcing detector and an adaptation of the LCG approach to exploit the transmission link diversities of users [9, 10]. In this part, we propose a new resource allocation scheme with MMSE detection technique for unirate multicast LP-OFDM systems. In multicast systems, when considering the LP-OFDM modulation technique, the loaded bits over the block S_b of subcarriers will be the lowest bit rate of users over this block.

To define the equivalent LP-OFDM channel, we need first to determine the distribution of subcarriers in each block. Let $\mathbf{D} = (d_{b,n})$ be a decision matrix of size $B \times N$. $\mathbf{D}$ determines the distribution of the N subcarriers into the B blocks and satisfies the following constraints:

$$d_{b,n} = \begin{cases} 1 & \text{if } n \in S_b \\ 0 & \text{else} \end{cases} \quad \forall n \in [1;N] \ \sum_{b=1}^{B} d_{b,n} = 1. \tag{21}$$

The optimal decision matrix $\mathbf{D}$ is such that the distortion is low for each block and is determined by solving an NP-hard combinatorial problem [10]. The proposed suboptimal definition of $\mathbf{D}$ consists in two steps. First, the subcarriers of the equivalent OFDM channel are sorted in descending order. Then, the adjacent subcarrier indices after the sorting operation are used to define the blocks. Let $\mathbf{O}$ be the vector of sorted indices of $|h_n^{\mathrm{eq}}|^2$ in descending order. The decision matrix is then

$$d_{b,n} = \begin{cases} 1 & \text{if } n \in \left\{O_j \ | (b-1)L + 1 \le j \le bL\right\}, \\ 0 & \text{else.} \end{cases} \tag{22}$$

Here is an example with $N = 8$, $L = 4$, $B = 2$, and $\mathbf{O} = [2\ 4\ 6\ 7\ 1\ 3\ 5\ 8]$. Hence, it follows that

$$\mathbf{D} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}. \tag{23}$$

For a given decision matrix $\mathbf{D}$, the equivalent LP-OFDM channel for the block b writes

$$\left| h_b^{\mathrm{eq}}(\mathbf{D}) \right|^2 = \min_k \frac{L}{\sum_{n=1}^{N} \left(d_{b,n} / \left| h_{k,n} \right|^2 \right)}. \tag{24}$$

4.3. Application of LP-OFDM Bit-Loading Algorithms. Based on the equivalent LP-OFDM channel, the multicast bit rate offered by the low block channel gain (LBCG) method proposed in [9] for ZF detector can then be rewritten as

$$\mathcal{R}_b^{\mathrm{LP}}(\mathbf{D}) = L \log_2 \left(1 + \frac{E}{\Gamma N_0} \left| h_b^{\mathrm{eq}}(\mathbf{D}) \right|^2 \right), \tag{25}$$

where $\mathbf{D}$ is defined in (22). The bit distributions $r_{u,b}$ for the different precoding sequences of the block are computed using (13). This method is called LBCG-ZF in reference to the detection technique used.

To determine the bit distribution with the MMSE detector for multicast LP-OFDM systems, Algorithm 1 is applied on the equivalent LP-OFDM channel (24). The newest method, which is based on MMSE detector, is called LBCG-MMSE.

5. Multirate Multicast Systems

Assuming that the multicast data are encoded into layers and any combination of the layers can be decoded at the receiver, the multicast bit rate can be increased by separating users according to their channel conditions. Under this assumption, the sender provides data in several layers organized in a hierarchy. Receivers subscribe to the layers cumulatively to provide progressive refinements [6–8]. If only the first layer is received by the user with the lowest data rate, the decoder produces the worst quality version. As more layers are received by more capable users, the decoder combines the layers to produce improved quality. To increase the total multicast bit rate, a first approach is based on the separation of users in frequency domain [3, 7]. Actually, each subcarrier is assigned to a subgroup of users which receive the same data symbols on this subcarrier. And then, the number of loaded bits on each subcarrier is determined considering the lowest one among the channel amplitudes of all the users allocated to this subcarrier. Notice that the subgroups of users are not the same for each subcarrier. This method, called frequency domain multirate multicast (FDMM) method, significantly increases the total multicast bit rate compared to the conventional LCG method in wireless communications [7].

It has been shown that this FDMM approach yields a good average bit rate, but the QoS requirements of users are not ensured and the fairness among users is degraded [9]. To enforce the fairness performance while minimizing throughput degradation, it has been proposed resource allocation algorithm with proportional fairness (PF) for multirate multicast OFDM systems [7]. In this part, we propose to jointly use the linear precoding technique with this PF-based approach to increase the multirate multicast bit rate. The proposed resource allocation algorithms take into account both ZF and MMSE detectors for multirate multicast LP-OFDM systems.

5.1. Bit Rate Optimization. The PF-based method adapts the bit rates of multicast users at each time slot according to previous allocated bit rates. Let $R_k(iT)$ be the bit rate of the kth user and r_n be the number of bits that are assigned to the nth subcarrier or the nth precoding sequence during the ith time slot of duration T. Let $\{r_{k,n}(iT)\}_{n\in[1;N]}$ be the bit distribution, resulting from bit-loading algorithms in OFDM case or LP-OFDM case, for the kth user in time slot i of duration T. For a low computational complexity, a simplified PF algorithm is developed by employing the average data rate, which is given by [21]

$$\begin{aligned} R_k(iT) = {} & \left(1 - \frac{1}{T_W} \right) R_k((i-1)T) \\ & + \frac{1}{T_W} \sum_{n=1}^{N} r_n(iT)\, \mathcal{U}\left(r_{k,n}(iT) - r_n(iT) \right), \end{aligned} \tag{26}$$

where T_W indicates the average window size and $\mathcal{U}$ the Heaviside step function, defined by

$$\mathcal{U}(x) = \begin{cases} 0 & \text{if } x < 0, \\ 1 & \text{if } x \geq 0. \end{cases} \tag{27}$$

The optimization problem, derived from [7], is written as

$$\max_{r_n} \sum_{k=1}^{K} R_k(iT) = \max_{r_n} \prod_{k=1}^{K} R_k(iT), \tag{28}$$

subject to given $R_k((i-1)T)$. As in [7], we show that the optimization problem (28) is asymptotically equivalent to the following one

$$\max_{r_n} \sum_{k=1}^{K} \sum_{n=1}^{N} \frac{r_n(iT)\, \mathcal{U}\left(r_{k,n}(iT) - r_n(iT) \right)}{R_k((i-1)T)}, \tag{29}$$

T_W; $\forall k, R_k(0) = 1$
for all user $k \in [1; K]$ **do**
  Compute $\{r_{k,n}\}_{n\in[1,N]}$ using ZF or MMSE detection bit-loading
end for
for all time slot index $i \in [1; T_W]$ **do**
  Select the user index k^* as defined in (30)
  Compute $\{r_n(iT)\}_{n\in[1,N]}$ using (29)
  Compute $R_k(iT)$ using (24)
end for

Algorithm 2: Bit-loading algorithm for multirate multicast systems.

subject to given $R_k((i-1)T)$. In fact, with $\mathcal{U}_{k,n} = \mathcal{U}(r_{k,n}(iT) - r_n(iT))$, the product in (28) is given by (30)

$$\begin{aligned}\prod_{k=1}^{K} R_k(iT) &= \underbrace{\left[\left(1-\frac{1}{T_W}\right)^K \prod_{k=1}^{K} R_k((i-1)T)\right]}_{=c_K \text{ (constant)}} \\ &\quad\times \prod_{k=1}^{K}\left(1+\frac{\sum_n r_n(iT)\mathcal{U}_{k,n}}{(T_W-1)R_k((i-1)T)}\right) \\ &= c_K\left[1+\frac{1}{T_W-1}\sum_{k=1}^{K}\sum_{n=1}^{N}\frac{r_n(iT)\mathcal{U}_{k,n}}{R_k((i-1)T)}\right. \\ &\quad+\left(\frac{1}{T_W-1}\right)^2 \\ &\quad\times\sum_{l\neq j}\frac{\sum_n r_n(iT)\mathcal{U}_{l,n}\sum_m r_m(iT)\mathcal{U}_{j,m}}{R_l((i-1)T)R_j((i-1)T)} \\ &\quad\left.+\cdots\right],\end{aligned} \tag{30}$$

and, for large T_W, the terms of orders greater than two can be neglected.

Since the considered Heaviside step function $\mathcal{U}$ in (27) is not continuous at zero, a local maximum cannot be found by using the first derivative test or second derivative test. A solution for the problem (29) is

$$r_n(iT) = r_{k^*,n}(iT), \tag{31}$$

where

$$k^* = \arg\max_k r_{k,n}(iT)\sum_{j=1}^{K}\frac{\mathcal{U}\left(r_{j,n}(iT) - r_{k,n}(iT)\right)}{R_j((i-1)T)}. \tag{32}$$

Algorithm 2 gives the corresponding bit-loading algorithm to provide the solution (31).

6. Results and Discussions

In this section, the performances of the resource allocation algorithms with the MMSE detector in multicast systems are presented. First, a performance comparison in terms of achieved total bit rate is realized for both unirate and multirate multicast systems. The different systems are summarized in Table 1. In unirate case, the LP-OFDM systems with both ZF and MMSE detection technique (LBCG-ZF and LBCG-MMSE) are compared with the conventional OFDM approach (LCG). In multirate case, the performance of the LP-OFDM systems (FDMM-LP-ZF and FDMM-LP-MMSE) are compared to the frequency domain multirate multicast approach for OFDM systems (FDMM-OFDM). Then, the fairness and complexities issues are presented.

Table 1: System characteristics and acronyms.

		Unirate	Multirate
OFDM-based		LCG	FDMM-OFDM
LP-based	ZF	LBCG-ZF	FDMM-LP-ZF
	MMSE	LBCG-MMSE	FDMM-LP-MMSE

The generated signal is composed of $N = 1024$ subcarriers transmitted in the (2; 27) MHz band. Perfect synchronization and channel estimation are assumed. A high background noise level of -110 dBm/Hz is considered, and the signal is transmitted with respect to a flat PSD of -50 dBm/Hz for all users. Results are given for a fixed target SER of 10^{-3} without channel coding. To determine the performances of the different algorithms, measured transfer functions of PLC channels are used. Here, the proposed multipath channel models for PLC in [22] are considered, where a classification of PLC channels is realized. PLC channels for indoor networks are classified into 9 classes per ascending order of their capacities, that is, the higher the channel class number, the better the channel amplitudes. A model of transfer function is associated to each class. Figure 4 shows three examples of PLC transfer function models. One channel of the category "good" (i.e., class 9 channel), one channel of the category "average" (i.e., class 5 channel), and one channel of the category "bad" (i.e., class 2 channel) are represented in the (2; 27) MHz band. In the following, a multicast system with a maximum of 9 users is considered, and each user experiences one different class of channel within the 9 classes.

6.1. Precoding Sequence Length Influence. We begin the simulation by highlighting the bit rate improvement provided

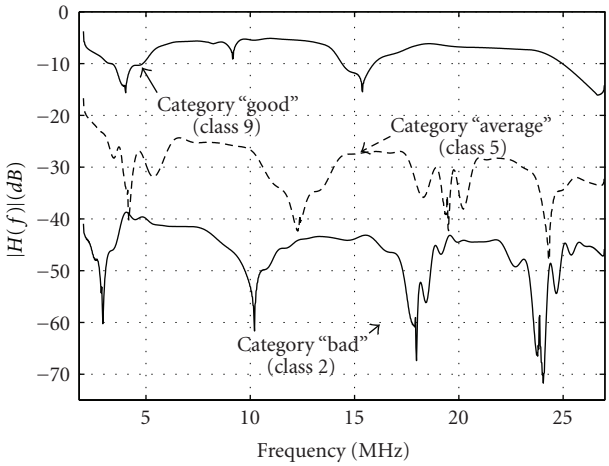


FIGURE 4: Transfer functions of three channel classes (class 2, class 5, and class 9 channels).

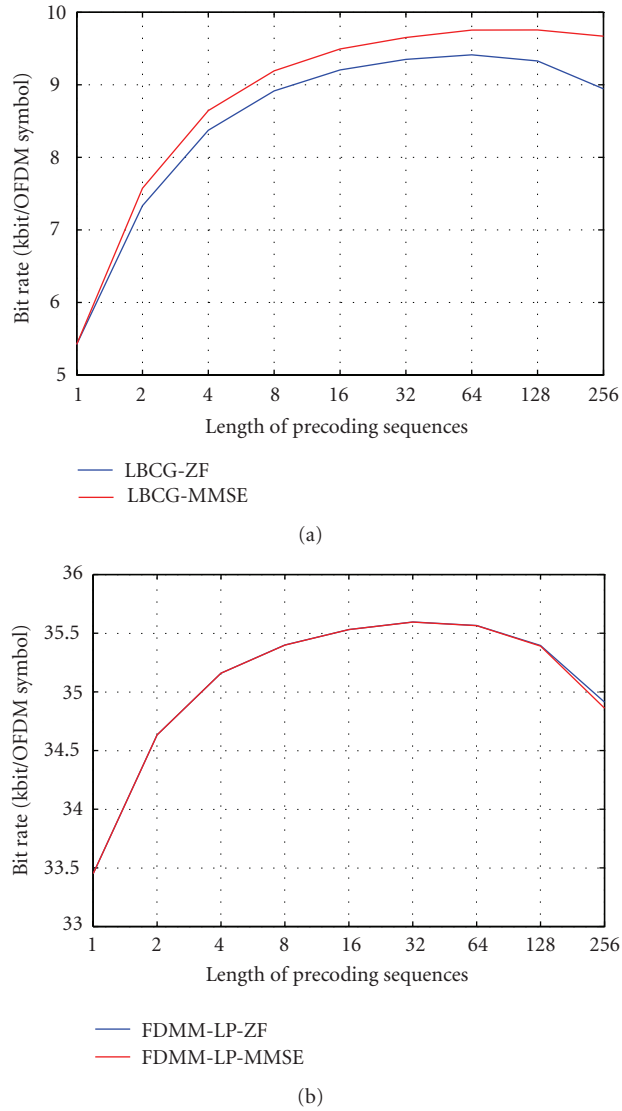
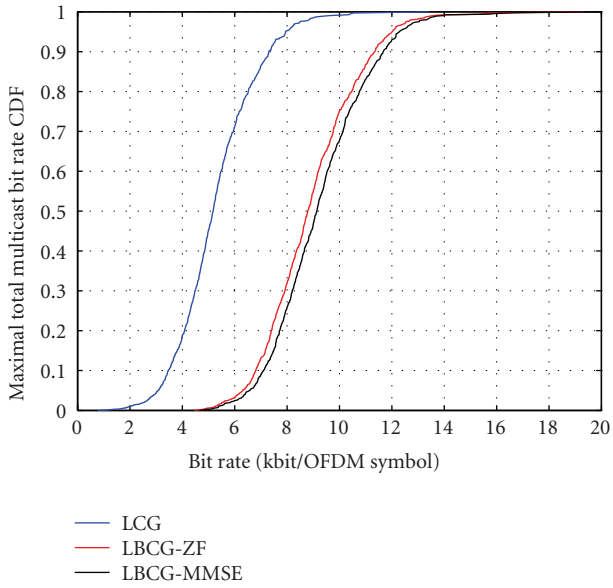


FIGURE 5: Total bit rate in bit per OFDM symbol versus the length of the precoding sequences for 9 users: (a) in unirate case and (b) in multirate case.

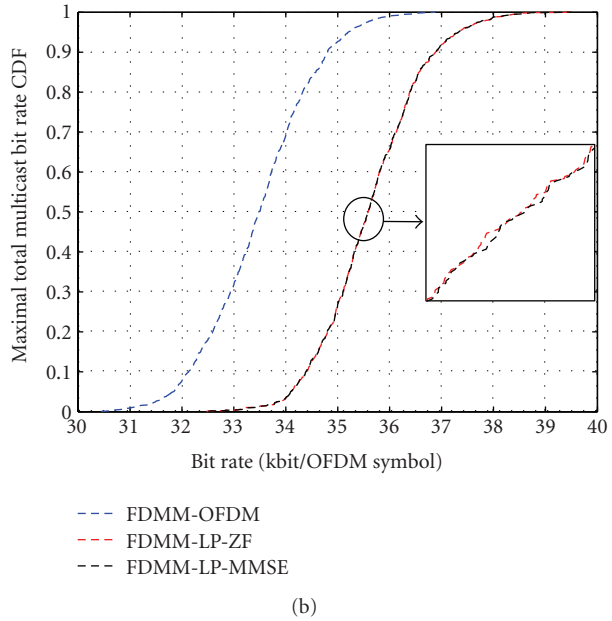
by the linear precoding component. For this purpose, we focus on results obtained when L varies. Notice that, finding the optimal precoding sequence lengths amounts to a complex combinatorial optimization problem that cannot be reduced to an equivalent convex problem. Thus, no analytical solution exists and optimal solution can only be obtained following exhaustive search [18]. Figure 5 shows the evolution of multicast LP-OFDM bit rate as a function of the length of the precoding sequences in (a) unirate case and (b) in multirate case. The parameter T_W is fixed to 10 to reduce the simulation time. It has been shown that the performance difference for small and large values of T_W is negligible [7]. The bit rate offered by the OFDM-based methods is given for $L = 1$. It is clear that the achievable bit rate with the LP-OFDM methods, whatever the equalization criteria, is improved when the length of the precoding sequences is greater than 1. This bit rate reaches a maximum value for $L = 64$ and $L = 32$, respectively in unirate and multirate multicast cases. In unirate case, the achieved multicast bit rate increases from approximately 5.42 kbit/OFDM symbol for LCG method with OFDM system to at least 9.35 kbit/OFDM symbol for LBCG methods with LP-OFDM system and ZF detector. In multirate case, this achieved bit rate increases from 33.35 kbit/OFDM symbol for FDMM-OFDM method to 35.5 kbit/OFDM symbol for FDMM-LP method with ZF and MMSE detectors. These improvements correspond to the bit rate gains of 42% and 6.4%, respectively for unirate and multirate multicast cases. Based on these results, we can state that the utilization of the linear precoding technique increases the bit rate of the multicast systems. The reason for the better performance of the LP-OFDM systems compared to OFDM systems is the efficient utilization of the PSD limit. The precoding component accumulates the residual energies of a given block of subcarriers to transmit additional bits.

In addition, the MMSE detection technique offers better performance than the ZF detection technique. The bit rate improvement reaches up to 8% in unirate multicast case, while the performance difference between ZF and MMSE detectors is minor in multirate multicast case. The low MMSE gain compared to the performance of ZF detector is due to the decision matrix $\mathbf{D}$ designed to reduce the distortion within each block. With low level of distortion, the powerfulness of the MMSE detector cannot be highlighted. Furthermore, the bit-loading algorithm for multirate multicast systems also reduces the distortion within each block to increase the multicast bit rate. The MMSE detector cannot improve with high gain the performance of the ZF detector.

6.2. Statistical Results of the Total Multicast Bit Rate. This part deals with the statistical results of the total multicast bit rate obtained through thousand simulations over a 9-user multicast system. They concern the cumulative distribution functions (CDF) of the maximal total multicast bit rates for $L = 32$ given in Figure 6. This CDF is also the bit rate



(a)



(b)

Figure 6: Cumulative distribution function (CDF) of the maximal total multicast bit rates: (a) in unirate case and (b) in multirate case for $L = 32$.

outage probability. Results confirm the better performances of the linear precoding technique in multicast systems. First, in unirate multicast case, the gap at 0.5 of the CDF between the maximal total bit rate of the LBCG and the LCG methods is at least 3.6 kbit/OFDM symbol, corresponding to a bit rate gain of 70%. In addition, the new resource allocation with MMSE detector (LBCG-MMSE) offers more bit rate than the LBCG-ZF method with low gains, around 3%. Second, in multirate multicast case, the gap at 0.5 of the CDF between the maximal total bit rate of the FDMM-LP and the FDMM-OFDM methods is at least 2.1 kbit/OFDM symbol,

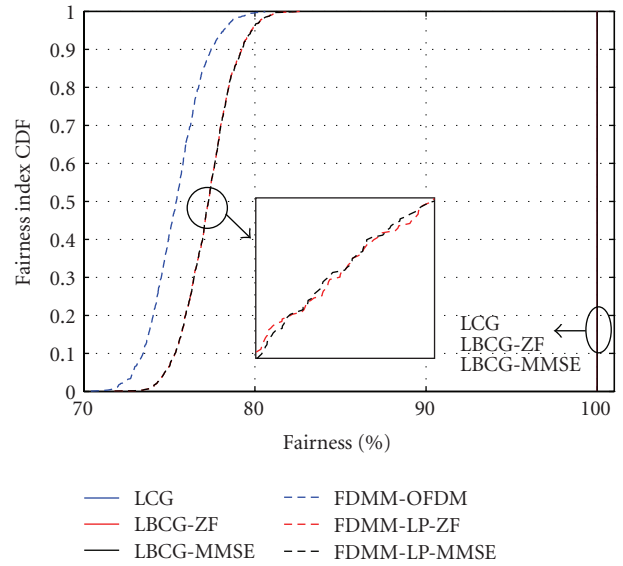


Figure 7: Cumulative distribution function of the fairness index of multicast users for $L = 32$.

corresponding to a bit rate gain of 6%. The performance difference between the FDMM-LP-ZF and the FDMM-LP-MMSE methods remain low in this context.

These results suggest that the linear precoding technique with MMSE detector is the best solution for resource allocation in unirate multicast systems. By cons, the linear precoding technique with ZF detector is sufficient in multirate multicast systems. Notice that the utilization of a more powerful equalizer such as minimum mean square error equalizer improves the LP-OFDM systems bit rate. However, the utilization of a ZF detector leads to fairly simple manipulations resolving the bit rate optimization problem in multirate context.

6.3. Fairness and Complexity Considerations. The unirate multicast methods equally distribute the resources, and all users receive the same bit rate. When the multicast users experience very different channel conditions, it is justifiable to give more resources to some users than others [10]. It is this idea that underlies the separation of users based on their channel conditions. The proposed algorithms in such a context have already demonstrated their better performance in terms of total multicast bit rate. It is also necessary to measure their performance in terms of fairness among users. As a performance metric, the fairness index defined in [23] is used as follows:

$$\mathrm{FI} = \frac{1}{U}\frac{\left(\sum_k R_k\right)^2}{\sum_k R_k^2}. \tag{33}$$

This fairness index measures the equality of user allocation and it is continuous so that any change in allocation changes the fairness also, contrary to the max-min fairness. Figure 7 shows the cumulative distribution function of the fairness index of multicast users. As expected, the unirate multicast

methods give the best fairness index, FI = 1. The LP-based methods, in addition to increase the total multicast bit rate, improve the fairness among the users. The fairness improvement is almost 2%.

In all cases, the MMSE detector outperforms the ZF detector. However, the ZF detector enhancement is low because the channel distortion has already been reduced by the decision matrix **D** and by the bit-loading algorithm for multirate multicast systems.

Besides the comparison of the performance in terms of bit rate and fairness index, the required downlink signaling overheads are compared. In unirate multicast systems, only the modulation order on each subcarrier or each precoding sequence needs to be signaled to users. In addition to information about the modulation order, the multirate multicast systems need to transmit information about the subgroups of users. Thus, it follows that the downlink signaling overhead of the FDMM based methods is higher than the other methods due to the subcarrier and bit allocation information. Furthermore, under the assumption that any combination of layers consisting of multicast data can be decoded at the receiver, an intelligent mapping algorithm for efficiently recovering the original data from different layers is needed [7]. This may bring additional signaling overhead.

7. Conclusion

In this paper, we have addressed the bit rate optimization problem in multicast linearly precoded OFDM systems in PLC context. A new resource allocation method with MMSE detector for multicast LP-OFDM systems has been proposed. The proposed method jointly uses linear precoded OFDM modulation technique and an adaptation of the OFDM-based multicast approaches to exploit the channel frequency selectivity experienced by each user in multicast OFDM systems. It has been shown through simulations that the proposed LP-based methods outperform the OFDM-based methods for both unirate and multirate multicast systems. Additionally, it is shown that the proposed bit-loading algorithm with MMSE detector offers the best performances in terms of total multicast bit rate and fairness among users.

Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under Grant agreement no. 213311 also referred to as OMEGA.

References

[1] B. Wang and J. Hou, "Multicast routing and its QoS extension: problems, algorithms, and protocols," *IEEE Journal of Network*, vol. 14, no. 1, pp. 22–36, 2000.

[2] S. Sadr, A. Anpalagan, and K. Raahemifar, "Radio resource allocation algorithms for the downlink of multiuser OFDM communication systems," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 3, pp. 92–106, 2009.

[3] C. Suh and C.-S. Hwang, "Dynamic subchannel and bit allocation for multicast OFDM systems," in *Proceedings of the IEEE 15th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '04)*, vol. 3, pp. 2102–2106, September 2004.

[4] J. Schmitt, F. Zdarsky, M. Karsten, and R. Steinmetz, "Heterogeneous multicast in heterogeneous QoS networks," in *Proceedings of the IEEE International Conference on Networks*, pp. 349–354, Bangkok, Thailand, October 2001.

[5] A. Mohamed and H. Alnuweiri, "Cross-layer optimal rate allocation for heterogeneous wireless multicast," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, Article ID 467182, 16 pages, 2009.

[6] N. Shacham, "Multicast routing of hierarchical data," in *Proceedings of the IEEE International Conference on Communications*, vol. 3, pp. 1217–1221, New York, NY, USA, March 1992.

[7] C. Suh and J. Mo, "Resource allocation for multicast services in multicarrier wireless communications," in *Proceedings of the IEEE International Conference on Computer Communications*, pp. 1–12, Barcelona, Spain, April 2006.

[8] C.-S. Hwang and Y. Kim, "An adaptive modulation method for multicast communications of hierarchical data in wireless networks," in *Proceedings of the IEEE International Conference on Communications*, pp. 896–900, New York, NY, USA, April 2002.

[9] A. Maiga, J.-Y. Baudais, and J.-F. Hélard, "Subcarrier, bit and time slot allocation for multicast precoded OFDM systems," in *Proceedings of the IEEE International Conference on Communications (ICC '10)*, pp. 1–6, Cap Town, South Africa, May 2010.

[10] A. Maiga, J.-Y. Baudais, O. Isson, A. Tonello, and S. D'Alessandro, "Optimized MAC algorithms and performance report," Tech. Rep., European OMEGA Project, 2010.

[11] A. Maiga, J.-Y. Baudais, and J.-F. Hélard, "Increase in multicast OFDM data rate in PLC network using adaptive LP-OFDM," in *Proceedings of the International Conference on Adaptive Science Technology*, pp. 384–389, Accra, Ghana, December 2009.

[12] J.-Y. Baudais, J.-F. Hélard, and J. Citerne, "An improved linear MMSE detection technique for multi-carrier CDMA systems: comparison and combination with interference cancellation schemes," *European Transactions on Telecommunications*, vol. 11, no. 6, pp. 547–554, 2000.

[13] S. Kaiser, "On the performance of different detection techniques for OFDM-CDMA in fading channels," in *Proceedings of the IEEE Global Telecommunications Conference*, pp. 2059–2063, November 1995.

[14] O. Isson, J.-M. Brossier, and D. Mestdagh, "Multi-carrier bit-rate improvement by carrier merging," *Electronics Letters*, vol. 38, no. 19, pp. 1134–1135, 2002.

[15] J.-Y. Baudais and M. Crussière, "Resource allocation with adaptive spread spectrum OFDM using 2D spreading for power line communications," *EURASIP Journal on Applied Signal Processing*, vol. 2007, Article ID 20542, 13 pages, 2007.

[16] S. Verdú, *Multiuser Detection*, Cambridge University Press, New York, NY, USA, 1998.

[17] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, New York, NY, USA, 2006.

[18] M. Crussière, J.-Y. Baudais, and J.-F. Hélard, "Robust and high-bit rate communications over PLC channels: a bit-loading multi-carrier spread-spectrum solution," in *Proceedings of the 9th International Symposium on Power Line*

Communications and Its Applications (ISPLC '05), pp. 37–41, Vancouver, Canada, April 2005.

[19] J. Cioffi, *A Multicarrier Primer*, Committee Contribution ANSI T1E1.4, 1991.

[20] C. H. Koh and Y. Y. Kim, "A proportional fair scheduling for multicast services in wireless cellular networks," in *Proceedings of the IEEE 64th Vehicular Technology Conference Fall (VTC '06)*, pp. 1–5, Montréal, Canada, September 2006.

[21] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system," in *Proceedings of the IEEE Vehicular Technology Conference Spring (VTC '02)*, vol. 3, pp. 1854–1858, May 2000.

[22] M. Tlich, A. Zeddam, F. Moulin, F. Gauthier, and G. Avril, "A broadband powerline channel generator," in *Proceedings of the IEEE International Symposium on Power Line Communications and Its Applications (ISPLC '07)*, pp. 505–510, Pisa, Italy, March 2007.

[23] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Computer Networks and ISDN Systems*, vol. 17, no. 1, pp. 1–14, 1989.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 680318, 12 pages
doi:10.1155/2012/680318

Research Article

Analytical Evaluation of the Performance of Proportional Fair Scheduling in OFDMA-Based Wireless Systems

Mohamed H. Ahmed, Octavia A. Dobre, and Rabie K. Almatarneh

Faculty of Engineering and Applied Science, Memorial University, St. John's, NL, Canada A1B 3X5

Correspondence should be addressed to Mohamed H. Ahmed, mhahmed@mun.ca

Received 2 March 2012; Accepted 7 May 2012

Academic Editor: Yi Su

This paper provides an analytical evaluation of the performance of proportional fair (PF) scheduling in Orthogonal Frequency-Division Multiple Access (OFDMA) wireless systems. OFDMA represents a promising multiple access scheme for transmission over wireless channels, as it combines the orthogonal frequency division multiplexing (OFDM) modulation and subcarrier allocation. On the other hand, the PF scheduling is an efficient resource allocation scheme with good fairness characteristics. Consequently, OFDMA with PF scheduling represents an attractive solution to deliver high data rate services to multiple users simultaneously with a high degree of fairness. We investigate a two-dimensional (time slot and frequency subcarrier) PF scheduling algorithm for OFDMA systems and evaluate its performance analytically and by simulations. We derive approximate closed-form expressions for the average throughput, throughput fairness index, and packet delay. Computer simulations are used for verification. The analytical results agree well with the results from simulations, which show the good accuracy of the analytical expressions.

1. Introduction

OFDMA is a promising solution for the high data-rate coverage required in multiuser broadband wireless communications. Current and evolving standards for broadband wireless systems, such as IEEE 802.16e, have proposed OFDMA as the multiple access technique for the air interface. OFDMA is a multiple access technique which is based on OFDM. In OFDM systems, a single user gets access to the whole available spectrum at any time instant, and, as a result, multiple users share resources using time scheduling. On the other hand, in OFDMA systems users share the available spectrum using subcarrier allocation. Hence, OFDMA requires scheduling in both time and frequency domains (time slots and frequency subcarriers). This additional degree of freedom makes the scheduling problem in OFDMA systems more challenging, but also more effective.

Scheduling plays a key role in the OFDMA systems resource management [1]. Efficient scheduling implies effective utilization of the available radio resources, high throughput, low packet delay, and fair treatment of all users in the system. Various scheduling techniques have been proposed for OFDMA systems [1–4]. For example, a maximum carrier-to-interference ratio-based scheduling algorithm is adopted in [1] to provide a more fair treatment among users, while in [2] the resource allocation problem is studied with and without service request constraints. Two-dimensional matrix-based scheduling algorithms are proposed in [2] using the raster scanning approach to achieve high system throughput with relatively lower complexity.

The PF algorithm is an appealing scheduling scheme to meet the quality of service requirements in OFDMA systems [5–8], as it can improve the fairness among users without sacrificing the efficiency in terms of average (or aggregate) throughput. With this algorithm, the level of satisfaction and starvation of all users in the system is sensed over time, and resources are assigned to users based on that. Moreover, the PF algorithm is flexible and can

scale between fairness and efficiency. In [8], we propose an iterative two-dimensional (time symbols and frequency subbands) PF scheduling for OFDMA systems. However, the performance of PF scheduling for OFDMA systems is not determined analytically and it is usually determined by computer simulations.

An analytical method, which is based on the Gaussian approximation of the instantaneous data rate in a Rayleigh fading environment, is used to analyze the performance of PF scheduling in [9]. However, this method is developed for single-carrier systems and limited to the case of users with full buffers. We adopt the methodology in [9] to develop an analytical solution for the PF scheduling in OFDMA systems for bursty traffic conditions and full buffers scenario, as well. In this paper, we provide approximate closed-form expressions for the average throughput and throughput fairness index of our PF scheduling scheme proposed for OFDMA systems in [8]. In addition, simulation results are provided in the paper to check the accuracy of the analytical method.

The rest of this paper is organized as follows: Section 2 describes the OFDMA system model. The PF scheduling algorithm is provided in Section 3. The closed-form analytical derivations of the throughput, fairness index, and delay are presented in Section 4. Then, Section 5 provides numerical results from the analytical solution, as well as simulation outcomes. Finally, conclusions are provided in Section 6.

2. System Model

As shown in Figure 1, the OFDMA system resources have two dimensions: frequency and time. In frequency domain, the signal bandwidth is divided into a plurality of subbands, which contain highly correlated orthogonal subcarriers. A number of S subcarriers are grouped into M subbands, each with $K = S/M$ subcarriers. In time domain, data is organized in frames, which are further divided in time symbols. The minimum allocable resource unit in the system is defined by the intersection between a subband in frequency domain and time symbol in time domain.

We consider a single-cell scenario, with N users with bursty traffic demands. The signals are affected by path loss, lognormal shadowing, and Rayleigh fading. The smallest data entity which the base station can handle is a fixed-size data packet. We use the Poisson traffic model. The cell shape is circular and the base station is located at the center. Users are uniformly distributed over the cell area. We consider the downlink only. However, the analysis can be easily extended to the uplink case. Moreover, adaptive coding and modulation (ACM) is used to enhance the resource utilization. The suitable modulation level and coding rate are decided depending on the channel state information (CSI) for each subband. Table 1 shows the ACM schemes used in this paper, along with the corresponding signal-to-noise ratios (SNRs).

The frequency subcarriers are correlated in the frequency domain. The fading affecting the frequency subcarriers has cross correlation because of the coherence bandwidth of the wireless channel [10]. A frequency selective Rayleigh fading channel is modeled based on [10–12]. The frequency selective Rayleigh subcarriers are generated with correlation between them in the frequency domain, where the complex valued correlation is formulated as a function of frequency separation between the subcarriers. In order to minimize the bit error rate and improve the OFDMA system reliability, we consider the worst case subcarrier fading in each subband for the SNR and link budget calculations. Although the worst case subcarrier fading is considered in a subband while selecting an ACM scheme, the overall SNR calculation does not significantly change because the fading difference between subcarriers within a subband is insignificant because the fading coefficients are highly correlated.

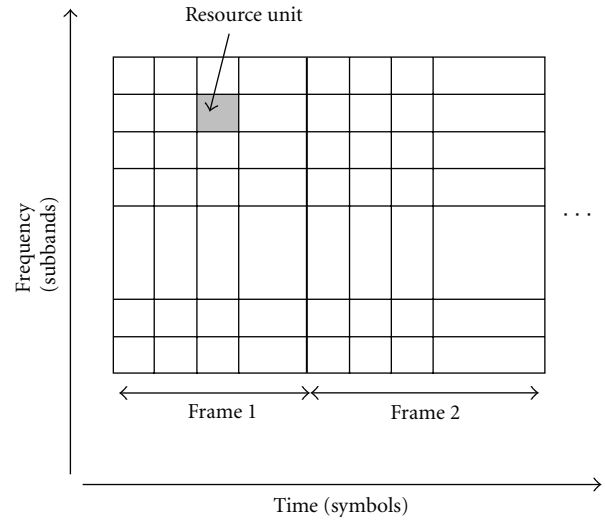


Figure 1: Two-dimensional resources in OFDMA systems.

Table 1: Adaptive coding and modulation parameters.

Modulation format	Code rate	Bits/symbol	SNR (dB)
BPSK	1/4	1/4	−2.9
BPSK	1/2	1/2	−0.2
QPSK	1/2	1	2.2
8PSK	1/2	3/2	5.2
8PSK	2/3	2	8.4
64QAM	1/2	3	11.8
64QAM	2/3	4	15.1

3. PF Scheduling Algorithm for OFDMA Systems

Closed-form expressions are subsequently derived for the throughput and fairness index for the PF scheduling algorithm that we proposed in [8]. The algorithm is briefly explained, followed by its analytical performance analysis.

According to the PF scheduling algorithm that we develop in [8] for OFDMA systems, the user with the index

$$k = \arg \max_{1 \le i \le N} \frac{D_{ij}(n)}{R_i(n-1)}, \tag{1}$$

is ranked first among the N users on subband j, $j = 1, \ldots, M$. Here, $D_{ij}(n)$ is the instantaneous data rate of user i, $i = 1, \ldots, N$ on subband j at time frame n, and $R_{i\backslash}(n)$ is the time-average data rate of user i at time frame n. The time-average data rate is updated at the end of a time frame for each user i on all the available subbands as follows:

$$R_i(n) = \begin{cases} (1 - T_c^{-1}) R_i(n-1), & i \neq k, \\ (1 - T_c^{-1}) R_i(n-1) + T_c^{-1} \sum\limits_{\substack{j=1, \\ j \in S_i(n)}}^{M} D_{ij}(n), & i = k, \end{cases} \tag{2}$$

where $S_i(n)$ represents the set of subbands assigned to user i during time frame n, and T_c is the averaging window expressed in time frames which controls the amount of historical information taken into account when sharing the resources among multiple users and can be chosen to achieve a desirable throughput-fairness tradeoff. User i is scheduled on time frame n if $i = k$ and is not scheduled if $i \neq k$.

Since the packet arrival is assumed to be bursty, the best user (chosen by (1)) might have empty buffer. In this case, the subband assigned to the best user should be given to the second best user if this has nonempty buffer. If not, the subband is assigned to the third best users and so on, where the ranking of users is based on the same criterion used in (1), that is, $D_{ij}(n)/R_{ij}(n-1)$. As such, we modify (2) as follows:

$$\begin{aligned} R_i(n) = {} & (1 - T_c^{-1}) R_i(n-1) \alpha T_c^{-1} \sum_{j=1}^{M} I_{ij}^{1}(n) D_{ij}(n) \\ & + \alpha(1-\alpha) T_c^{-1} \sum_{j=1}^{M} I_{ij}^{2}(n) D_{ij}(n) \\ & + \alpha(1-\alpha)^2 T_c^{-1} \sum_{j=1}^{M} I_{ij}^{3}(n) D_{ij}(n) \\ & + \cdots + \alpha(1-\alpha)^{N-1} T_c^{-1} \sum_{j=1}^{M} I_{ij}^{N}(n) D_{ij}(n) \\ = {} & (1 - T_c^{-1}) R_i(n-1) + \alpha T_c^{-1} \sum_{k=1}^{N} (1-\alpha)^{k-1} \\ & \times \sum_{j=1}^{M} I_{ij}^{k}(n) D_{ij}(n), \end{aligned} \tag{3}$$

where $I_{ij}^{k}(n)$, $k = 1, \ldots, N$, represents a selector indicator which equals 1 if user i is ranked kth on subband j and frame n and equals 0 otherwise, and α is the probability that the buffer of user i is not empty. We assume that α is the same for all users. The terms in the right-hand side of (3) represent the potential achievable throughput for a user. The first term reflects the average throughput achieved by the round-robin (RR) algorithm, while the remaining N terms represent the additional average throughput provided by our algorithm when compared with RR. The first term (out of the remaining N terms) represents the additional average throughput when user i is ranked first and assigned subband j. The second term (out of the remaining N terms) reflects the additional average throughput when user i is ranked second and assigned subband j because the user ranked first has empty buffer, and so on.

The PF scheduling algorithm consists of two steps [8]. In the first step, all users in the system are ranked. A resource matrix that contains the ranking of all users on all subbands is generated based on (1). The instantaneous data rate, $D_{ij}(n)$, represents the efficiency factor, whereas the historical average rate combined with T_c represents the fairness factor. As such, the ranking of the users reflects both the channel gain and shortage of service. In the second step, scheduling is performed based on the ranking and demands of the users on one hand and the resource accessibility on the other hand. The algorithm iteratively serves the user with the highest rank among all users on all subbands.

A user will be excluded from the waiting users' list if all waiting packets are served. This algorithm allows subband sharing in time domain, where different time symbols in the subband can be utilized by different users. A subband will be eliminated from the resource matrix if the remaining resources cannot support at least one packet for any requesting user within this time frame. The algorithm tracks the satisfaction levels of all users at the end of each time frame by updating the historical data rate, $R_i(n)$, using (2).

4. Performance Analysis

4.1. Average Throughput. It is shown that assuming a linear relationship between the instantaneous data rate, $D_{ij}(n)$, and the SNR is unrealistic under Rayleigh fading environment [9, 13]. Actually, it is demonstrated that it is more realistic to assume that $D_{ij}(n)$ follows a Gaussian distribution with mean and variance given, respectively, as follows [9]:

$$\begin{aligned} E\left[D_{ij}\right] &= \int_0^{\infty} \log\left(1 + \mathrm{SNR}_{ij}\gamma\right) e^{-\gamma} d\gamma, \\ \sigma_{D_{ij}}^2 &= \int_0^{\infty} \log\left(1 + \mathrm{SNR}_{ij}\gamma\right)^2 e^{-\gamma} d\gamma \\ &\quad - \left(\int_0^{\infty} \log\left(1 + \mathrm{SNR}_{ij}\gamma\right) e^{-\gamma} d\gamma \right)^2, \end{aligned} \tag{4}$$

where $E[\cdot]$ denotes the expectation operator. According to the PF algorithm presented in (1) and (2), one can express

the average achievable throughput of user i on all the available subbands in the time frame n as follows:

$$E[R_i(n)] = \left(1 - T_c^{-1}\right)E[R_i(n-1)] + \alpha T_c^{-1}\sum_{k=1}^{N}(1-\alpha)^{k-1}E\left[\sum_{j=1}^{M} I_{ij}^{k}(n)D_{ij}(n)\right]. \quad (5)$$

We can rewrite (5) as follows:

$$E[R_i(n)] = \left(1 - T_c^{-1}\right)E[R_i(n-1)] + \alpha T_c^{-1}\sum_{k=1}^{N}(1-\alpha)^{k-1} \times E\left[\sum_{j=1}^{M} D_{ij}(n) \mid I_{ij}^{k}(n) = 1\right]\Pr\left(I_{ij}^{k}(n) = 1\right), \quad (6)$$

where $\Pr(I_{ij}^{k}(n) = 1)$ is the probability that user i is ranked kth on subband j and time frame n. Under the assumption of stationary throughput [9], R_i, and independent subbands, one can further express (6) as follows:

$$E[R_i] = \alpha\sum_{k=1}^{N}(1-\alpha)^{k-1} \times E\left[\sum_{j=1}^{M} D_{ij}(n) \mid I_{ij}^{k}(n) = 1\right]\Pr\left(I_{ij}^{k}(n) = 1\right). \quad (7)$$

By applying the Bayes' theorem, (7) can be rewritten as follows:

$$E[R_i] = \alpha\sum_{k=1}^{N}(1-\alpha)^{k-1} \times \sum_{j=1}^{M}\int_{-\infty}^{\infty} x f_{D_{ij}}(x)\Pr\left(I_{ij}^{k}(n) = 1 \mid D_{ij}(n) = x\right)dx, \quad (8)$$

where $f_{D_{ij}}(\cdot)$ denotes the probability density function (pdf) of D_{ij}. By assuming independent D_{ij} and based on the PF selection criterion presented in (1), we can determine the conditional ranking probabilities as follows:

$$\Pr\left(I_{ij}^{1}(n) = 1 \mid D_{ij}(n) = x\right) = \prod_{\substack{l=1\\ l\neq i}}^{N} F_{D_{lj}}\left(x\frac{R_l(n)}{R_i(n)}\right),$$

$$\Pr\left(I_{ij}^{2}(n) = 1 \mid D_{ij}(n) = x\right) = \left(1 - F_{D_{l_1 j}}\left(x\frac{R_{l_1}(n)}{R_i(n)}\right)\right)\prod_{\substack{l=1\\ l\neq i,l_1}}^{N} F_{D_{lj}}\left(x\frac{R_l(n)}{R_i(n)}\right),$$

$$\Pr\left(I_{ij}^{3}(n) = 1 \mid D_{ij}(n) = x\right) = \left(1 - F_{D_{l_1 j}}\left(x\frac{R_{l_1}(n)}{R_i(n)}\right)\right)\left(1 - F_{D_{l_2 j}}\left(x\frac{R_{l_2}(n)}{R_i(n)}\right)\right) \times \prod_{\substack{l=1\\ l\neq i,l_1,l_2}}^{N} F_{D_{lj}}\left(x\frac{R_l(n)}{R_i(n)}\right),$$

$$\vdots$$

$$\Pr\left(I_{ij}^{N}(n) = 1 \mid D_{ij}(n) = x\right) = \prod_{\substack{l=1\\ l\neq i}}^{N}\left(1 - F_{D_{lj}}\left(x\frac{R_l(n)}{R_i(n)}\right)\right), \quad (9)$$

where $F_{D_{ij}}(\cdot)$ is the cumulative distribution function (cdf) of D_{ij}, while l_1 and l_2 are the indexes of the users ranked the first and the second (on subband j), respectively. By using (9) and the Gaussian pdf of D_{ij}, and under the assumptions that $T_c \to \infty$ and R_i is an ergodic process (such that its moving average equals the statistical average), now (9) can be re-written as follows:

$$\Pr\left(I_{ij}^{1}(n) = 1 \mid D_{ij}(n) = x\right) \approx \prod_{\substack{l=1\\ l\neq i}}^{N} F_{D_{lj}(n)}\left(\frac{E[R_l]}{E[R_i]}x\right),$$

$$\Pr\left(I_{ij}^{2}(n) = 1 \mid D_{ij}(n) = x\right) \approx \left(1 - F_{D_{ij}(n)}\left(\frac{E[R_l]}{E[R_i]}x\right)\right) \times \prod_{\substack{l=1\\ l\neq i}}^{N-1} F_{D_{lj}(n)}\left(\frac{E[R_l]}{E[R_i]}x\right)$$

$$\Pr\left(I_{ij}^{3}(n) = 1 \mid D_{ij}(n) = x\right) \approx \left(1 - F_{D_{ij}(n)}\left(\frac{E[R_l]}{E[R_i]}x\right)\right)^{2}\prod_{\substack{l=1\\ l\neq i}}^{2N-2} F_{D_{lj}(n)}\left(\frac{E[R_l]}{E[R_i]}x\right),$$

$$\vdots$$

$$\Pr\left(I_{ij}^{N}(n) = 1 \mid D_{ij}(n) = x\right) = \prod_{\substack{l=1\\ l\neq i}}^{N}\left(1 - F_{D_{lj}}\left(\frac{E[R_l]}{E[R_i]}x\right)\right). \quad (10)$$

Hence, (8) can be expressed as follows:

$$E[R_i] = \alpha \sum_{k=1}^{N} (1-\alpha)^{k-1} \times \sum_{j=1}^{M} \left[\int_{-\infty}^{\infty} x f_{D_{ij}(n)}(x) \left[1 - F_{R_{ij}(n)} \left(\frac{E[R_l]}{E[R_i]} x \right) \right]^{k-1} \times \prod_{\substack{l=1 \\ l \neq i}}^{N-k+1} F_{D_{lj}(n)} \left(\frac{E[R_l]}{E[R_i]} \right) dx \right]. \tag{11}$$

By assuming a Gaussian distribution of the instantaneous traffic rate, (11) becomes

$$E[R_i] = \alpha \sum_{k=1}^{N} (1-\alpha)^{k-1} \times \sum_{j=1}^{M} \left[\int_{-\infty}^{\infty} \left(y\sigma_{D_{ij}} + E\left[D_{ij}\right] \right) \frac{e^{-y^2/2}}{\sqrt{2\pi}} \times \left[1 - F_{R_{ij}(n)} \left(\frac{E[R_l]}{E[R_i]} \left(y\sigma_{D_{ij}} + E\left[D_{ij}\right] \right) \right) \right]^{k-1} \times \prod_{\substack{l=1, \\ l \neq i}}^{N-k+1} F_{D_{lj}(n)} \left(\frac{E[R_l]}{E[R_i]} \left(y\sigma_{D_{ij}} + E\left[D_{ij}\right] \right) \right) dy \right]. \tag{12}$$

Now, assume $E[R_l]/E[R_i]) = (E[D_l]/E[D_i]$, so, $F_{R_{ij}(n)}(y\sigma_{D_{ij}} + E[D_{ij}])$ can be re-written as [8]

$$F_{R_{ij}(n)} \left(\frac{E[R_l]}{E[R_i]} \left(y\sigma_{D_{ij}} + E\left[D_{ij}\right] \right) \right) = F_{(0,1)} \left(\frac{E[D_{li}]\sigma_{D_{li}}}{E\left[R_{ij}\right]\sigma_{D_{ij}}} y \right), \tag{13}$$

where $F_{(0,1)}(\cdot)$ represents the standard normal cdf with zero-mean and unit-variance. Furthermore, we assume a proportional relationship between the mean and standard deviation of all users in the system [8]; hence, the previous expression can be approximated as

$$F_{(0,1)} \left(\frac{E[D_{li}]\sigma_{D_{li}}}{E\left[R_{ij}\right]\sigma_{D_{ij}}} y \right) = F_{(0,1)}(y). \tag{14}$$

After some mathematical manipulations, one can further express (12) as

$$E[R_i] = \alpha \sum_{k=1}^{N} (1-\alpha)^{k-1} \times \sum_{j=1}^{M} \left[\sigma_{D_{ij}} \int_{-\infty}^{\infty} y \frac{e^{-y^2/2}}{\sqrt{2\pi}} \left(1 - F_{(0,1)}(y)\right)^{k-1} F_{(0,1)}^{N-k}(y) dy + E\left[D_{\mathrm{ij}}\right] \int_{0}^{1} \left(1 - F_{(0,1)}(y)\right)^{k-1} F_{(0,1)}^{N-k}(y) dF_{(0,1)}(y) \right]. \tag{15}$$

It is straightforward to show that

$$\int_{0}^{1} F_{(0,1)}^{N-1}(y) dF_{(0,1)}(y) = \frac{1}{N}. \tag{16}$$

Then, one can easily find that

$$\int_{0}^{1} \left(1 - F_{(0,1)}(y)\right) F_{(0,1)}^{N-2}(y) dF_{(0,1)}(y) = \frac{1}{N(N-1)}, \tag{17}$$

and, finally, through the mathematical induction, we can write

$$\int_{0}^{1} \left(1 - F_{(0,1)}(y)\right)^{k-1} F_{(0,1)}^{N-k}(y) dF_{(0,1)}(y) = \frac{(k-1)!(N-k)!}{N!}, \quad k = 1, \ldots, N. \tag{18}$$

Thus, (15) can be expressed as follows:

$$E[R_i] = \alpha \sum_{k=1}^{N} (1-\alpha)^{k-1} \times \sum_{j=1}^{M} \sigma_{D_{ij}} \int_{-\infty}^{\infty} y \frac{e^{-y^2/2}}{\sqrt{2\pi}} \left(1 - F_{(0,1)}(y)\right)^{k-1} \times F_{(0,1)}^{N-k}(y) dy + E\left[D_{ij}\right] \frac{(k-1)!(N-k)!}{N!}. \tag{19}$$

The probability of the nonempty buffer for any user, α, in terms of average throughput and traffic rate, is given as follows:

$$\alpha = \frac{\lambda}{E[R_i]}, \tag{20}$$

where λ is the average arrival traffic rate per user. By substituting (20) into (19), $E[R_i]$ becomes

$$E[R_i] = \sum_{k=1}^{N} \frac{\lambda}{E[R_i]} \left(1 - \frac{\lambda}{E[R_i]}\right)^{k-1} \times \sum_{j=1}^{M} \sigma_{D_{ij}} \int_{-\infty}^{\infty} y \frac{e^{-y^2/2}}{\sqrt{2\pi}} \left(1 - F_{(0,1)}(y)\right)^{k-1} F_{(0,1)}^{N-k}(y) dy + E\left[D_{ij}\right] \frac{(k-1)!(N-k)!}{N!}. \tag{21}$$

As $E[R_i]$ represents the throughput of user i in the system, the average throughput of the entire system is

$$E[R] = \sum_{i=1}^{N} E[R_i]. \tag{22}$$

4.2. Fairness Index. Jain's fairness index is a well-known quantitative metric that is widely used in wireless communications to measure fairness, and it is defined as follows [14]:

$$J(x_1, x_2, x_3, \ldots, x_N) = \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N \sum_{i=1}^{N} x_i^2}, \tag{23}$$

where x_i is the amount of resources accessed by user i among N competing users. Based on the result for the average throughput for user i, as given in (21), it is straightforward to express the Jain's fairness index of the users' throughput as follows:

$$J(E[R_1], E[R_2], E[R_3], \ldots, E[R_N]) = \frac{\left(\sum_{i=1}^{N} E[R_i]\right)^2}{\sum_{i=1}^{N} E[R_i]^2}. \tag{24}$$

For nonbursty traffic (full-buffer scenario), the analysis is the same as for bursty traffic given above, except that α (the probability of having non-empty buffer) is equal to 1.

4.3. Average Packet Delay. In order to calculate the packet delay, we model the system by using the M/G/1 queuing model. Hence, the average packet delay is given by

$$\omega_i = \frac{1}{E[R_i]} + \frac{\lambda_i\left(1/\left(E^2[R_i] + \sigma_{R_i}^2\right)\right)}{2(1 - (\lambda_i / E[R_i]))}, \tag{25}$$

where $\sigma_{R_i}^2$ is the throughput variance. In order to determine $\sigma_{R_i}^2$, we calculate $E[R_i^2(n)]$ using (3) as follows:

$$\begin{aligned} E[R_i^2(n)] = {} & \left(\frac{T_c - 1}{T_c}\right)^2 E[R_i^2(n-1)] \\ & + \frac{1}{T_c^2} \sum_{i=1}^{N} \alpha^2 (1-\alpha)^{2(i-1)} E\left[\sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)\right]^2 \\ & + \frac{2(T_c - 1)}{T_c^2} E[R_i(n-1)] \\ & \times E\left[\sum_{i=1}^{N} \alpha (1-\alpha)^{i-1} \sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)\right]. \end{aligned} \tag{26}$$

By assuming stationary throughput per user, we can use $E[R_i(n)] = E[R_i(n-1)]$. Therefore, (26) can be re-written as follows:

$$\begin{aligned} (2T_c - 1) E[R_i^2] = {} & \sum_{i=1}^{N} \alpha^2 (1-\alpha)^{2(i-1)} E \\ & \times \left[\sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)\right]^2 + 2(T_c - 1) \\ & \times E[R_i] E\left[\sum_{i=1}^{N} \alpha (1-\alpha)^{i-1} \sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)\right]. \end{aligned} \tag{27}$$

In order to determine $E[R_i^2]$, we need to find $E[\sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)]^2$, which can be expressed as follows:

$$E\left[\sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)\right]^2 = \sum_{j=1}^{M} E\left(D_{ij}^2 I_{ij}^i\right) + \sum_{j=1}^{M} \sum_{\substack{h=1, \\ h \neq j}}^{M} E\left(D_{ij} D_{ih} I_{ij}^i I_{ih}^i\right), \tag{28}$$

and then can be re-written as

$$\begin{aligned} & E\left[\sum_{j=1}^{M} D_{ij}(n) I_{ij}^i(n)\right]^2 \\ & = \sum_{j=1}^{M} \Pr\left(I_{ij}^i = 1\right) \\ & \quad \times \int_{-\infty}^{\infty} x^2 f_{D_{ij}}\left(x \mid I_{ij}^i = 1\right) dx \\ & \quad + \sum_{j=1}^{M} \Pr\left(I_{ij}^i = 1\right) \left(\int_{-\infty}^{\infty} x f_{D_{ij}}\left(x \mid I_{ij}^i = 1\right) dx\right) \\ & \quad \times \sum_{\substack{h=1, \\ h \neq j}}^{M} \Pr\left(I_{ih}^i = 1\right) \left(\int_{-\infty}^{\infty} x f_{D_{ih}}\left(x \mid I_{ih}^i = 1\right) dx\right). \end{aligned} \tag{29}$$

The first term in the right-hand side of (29) can be further written as follows:

$$\begin{aligned} & \sum_{j=1}^{M} \Pr\left(I_{ij}^i = 1\right) \int_{-\infty}^{\infty} x^2 f_{D_{ij}}\left(x \mid I_{ij}^i = 1\right) dx \\ & \quad = \sum_{j=1}^{M} \int_{-\infty}^{\infty} x^2 f_{D_{ij}}(x) \Pr\left(I_{ij}^i = 1 \mid D_{ij} = x\right) dx. \end{aligned} \tag{30}$$

Using (9) and the assumption of stationary first-order ergodic R_i [9], (30) becomes

$$\sum_{j=1}^{M}\Pr\left(I_{ij}^{i}=1\right)\int_{-\infty}^{\infty}x^2 f_{D_{ij}}\left(x\mid I_{ij}^{i}=1\right)dx = \sum_{j=1}^{M}\int_{-\infty}^{\infty}x^2 f_{D_{ij}}(x)\left(1-F_{D_{ij}}\left(\frac{E[R_l(n)]}{E[R_i(n)]}x\right)\right)^{i-1} \times\prod_{\substack{l=1,\\ l\neq i}}^{N-i}F_{D_{ij}}\left(\frac{E[R_l(n)]}{E[R_i(n)]}x\right)dx, \tag{31}$$

which can be simplified to

$$\sum_{j=1}^{M}\Pr\left(I_{ij}^{i}=1\right)\int_{-\infty}^{\infty}x^2 f_{D_{ij}}\left(x\mid I_{ij}^{i}=1\right)dx = \sum_{j=1}^{M}\int_{-\infty}^{\infty}\left(y\sigma_{D_{ij}}+E\left[D_{ij}\right]\right)^2 f_{D_{ij}}(y) \times\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx. \tag{32}$$

Then, by simply expressing $(y\sigma_{D_{ij}}+E[D_{ij}])^2$, (32) can be re-written as follows:

$$\begin{aligned}\sum_{j=1}^{M}\Pr\left(I_{ij}^{i}=1\right)\int_{-\infty}^{\infty}x^2 f_{D_{ij}}\left(x\mid I_{ij}^{i}=1\right)dx &= \sum_{j=1}^{M}\sigma_{D_{ij}}^2\int_{-\infty}^{\infty}y^2 f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx\\ &+2\sigma_{D_{ij}}E\left[D_{ij}\right]\int_{-\infty}^{\infty}y f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx\\ &+E\left[D_{ij}\right]^2\int_{-\infty}^{1}y^2 f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx.\end{aligned} \tag{33}$$

Thus, $\sum_{j=1}^{M}\Pr(I_{ij}^{i}=1)\int_{-\infty}^{\infty}x^2 f_{D_{ij}}(x\mid I_{ij}^{i}=1)dx$ can be expressed as follows:

$$\begin{aligned}\sum_{j=1}^{M}\Pr\left(I_{ij}^{i}=1\right)\int_{-\infty}^{\infty}x^2 f_{D_{ij}}\left(x\mid I_{ij}^{i}=1\right)dx &= \sum_{j=1}^{M}\sigma_{D_{ij}}^2\int_{-\infty}^{\infty}y^2 f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx\\ &+2\sigma_{D_{ij}}E\left[D_{ij}\right]\int_{-\infty}^{\infty}y f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx\\ &+\frac{(i-1)E\left[D_{ij}\right]}{N!/(N-i)!}.\end{aligned} \tag{34}$$

Next, we determine the second term in the right-hand side of (29), which can be re-written as follows:

$$\begin{aligned}&\sum_{j=1}^{M}\Pr\left(I_{ij}^{i}=1\right)\left(\int_{-\infty}^{\infty}x f_{D_{ij}}\left(x\mid I_{ij}^{i}=1\right)dx\right)\\ &\quad\times\sum_{\substack{h=1,\\ h\neq j}}^{M}\Pr\left(I_{ik}^{i}=1\right)\left(\int_{-\infty}^{\infty}x f_{D_{ih}}\left(x\mid I_{ih}^{i}=1\right)dx\right)\\ &=\sum_{j=1}^{M}\left(\sigma_{D_{ij}}\int_{-\infty}^{\infty}\frac{ye^{-y^2/2}}{\sqrt{2\pi}}\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}(y)^{N-i}dy+\frac{(i-1)E\left[D_{ij}\right]}{N!/(N-i)!}\right)\\ &\quad\times\sum_{\substack{h=1,\\ h\neq j}}^{M}\left(\sigma_{D_{ik}}\int_{-\infty}^{\infty}\frac{ye^{-y^2/2}}{\sqrt{2\pi}}\left(1-F_{(0,1)}(y)\right)^{i-1}\times F_{(0,1)}(y)^{N-i}dy+\frac{(i-1)E[D_{ih}]}{N!/(N-i)!}\right).\end{aligned} \tag{35}$$

From (29), (34) and (35), $E[\sum_{j=1}^{M}D_{ij}(n)I_{ij}^{i}(n)]^2$ can be expressed as follows:

$$\begin{aligned}E\left[\sum_{j=1}^{M}D_{ij}(n)I_{ij}^{i}(n)\right]^2 &=\sum_{j=1}^{M}\sigma_{D_{ij}}^2\int_{-\infty}^{\infty}y^2 f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx\\ &+2\sigma_{D_{ij}}E\left[D_{ij}\right]\int_{-\infty}^{\infty}y f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)dx\\ &+\frac{(i-1)E\left[D_{ij}\right]}{N!/(N-i)!}\\ &+\sum_{j=1}^{M}\left(\sigma_{D_{ij}}\int_{-\infty}^{\infty}\frac{ye^{-y^2/2}}{\sqrt{2\pi}}\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}(y)^{N-i}dy+\frac{(i-1)E\left[D_{ij}\right]}{N!/(N-i)!}\right)\\ &\times\sum_{\substack{h=1,\\ h\neq j}}^{M}\left(\sigma_{D_{ik}}\int_{-\infty}^{\infty}\frac{ye^{-y^2/2}}{\sqrt{2\pi}}\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}(y)^{N-i}dy+\frac{(i-1)E[D_{ih}]}{N!/(N-i)!}\right)\end{aligned} \tag{36}$$

Then, we simplify the second term in the right-hand side of (27) as follows:

$$2(T_c-1)E[R_i]E\left[\sum_{i=1}^{N}\alpha(1-\alpha)^{i-1}\sum_{j=1}^{M}D_{ij}(n)I_{ij}^{i}(n)\right] = 2(T_c-1)E[R_i]E[R_i] = 2(T_c-1)E[R_i]^2. \quad (37)$$

Substituting (36) and (37) in (27), it can be easily shown that the throughput variance is expressed as:

$$\begin{aligned}\sigma_{R_i}^2 &= E[R_i^2]-E[R_i]^2\\ &= \frac{1}{2(T_c-1)}\left(\sum_{i=1}^{N}\alpha^2(1-\alpha)^{2(i-1)}\right.\\ &\quad\times\sum_{j=1}^{M}\sigma_{D_{ij}}^2\int_{-\infty}^{\infty}y^2 f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)\,dx\\ &\quad+2\sigma_{D_{ij}}E\left[D_{ij}\right]\int_{-\infty}^{\infty}y f_{D_{ij}}(y)\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)\,dy\\ &\quad+\frac{(i-1)E\left[D_{ij}\right]}{N!/(N-i)!}\sum_{i=1}^{N}\alpha^2(1-\alpha)^{2(i-1)}\\ &\quad\times\sum_{j=1}^{M}\left(\sigma_{D_{ij}}\int_{-\infty}^{\infty}\frac{ye^{-y^2/2}}{\sqrt{2\pi}}\left(1-F_{(0,1)}(y)\right)^{i-1}F_{(0,1)}^{N-i}(y)\,dy\right.\\ &\qquad\left.+\frac{(i-1)E\left[D_{ij}\right]}{N!/(N-i)!}\right)\\ &\quad\times\sum_{\substack{k=1,\\k\neq j}}^{M}\left(\sigma_{D_{ik}}\int_{-\infty}^{\infty}\frac{ye^{-y^2/2}}{\sqrt{2\pi}}\left(1-F_{(0,1)}(y)\right)^{i-1}\right.\\ &\qquad\left.\left.\times F_{(0,1)}^{N-i}(y)\,dy+\frac{(i-1)E[D_{ik}]}{N!/(N-i)!}\right)\right).\end{aligned} \quad (38)$$

By substituting (21) and (38) in (25), we can calculate the average packet delay (ω_i).

5. Numerical and Simulation Results

The accuracy of the analytical closed-form expressions for the average throughput, fairness index, and packet delay (derived in Section 4) is examined by comparing the analytical results with simulation results. Computer simulations of one cell with N users are conducted independently of the analytical expressions derived in the previous section to estimate the average throughput, fairness index, and packet delay. We set the signal bandwidth to 20 MHz, the carrier frequency to 2 GHz, the noise power to −130 dBW, and T_c to 5000 frames (except in Figures 2, 3, and 10). In addition, we consider a path loss exponent of 4, the standard deviation of the lognormal shadowing equal to 10 dB, the cell radius set to 1500 m, the number of users, N, in the cell equal to 32, the frame duration of 2 ms, and the packet size of 180 bits. The number of subbands, M, is 32 and the number of subcarriers, S, is 256. We use Poisson traffic with an arrival rate of λ, which is kept as a variable to control the traffic load given by λN.

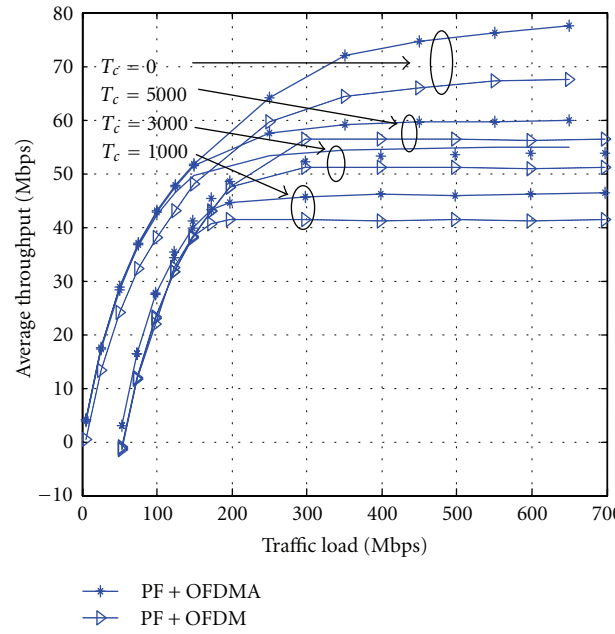


Figure 2: Average throughput of the PF with OFDMA and PF with OFDM.

We first analyze the effect of the averaging window (T_c) and the impact of using OFDMA instead of OFDM. In OFDM, all subcarriers are given to the selected user by the PF. As shown in Figure 2 (when $T_c > 0$) the larger the T_c the higher the throughput. When T_c increases, PF needs more time to compensate disadvantaged users (with low SNR), which leads to a higher throughput for the advantaged users (with good SNR). As a result, the average throughput increases. On the other hand, when $T_c = 0$, PF losses its fairness and becomes an opportunistic scheduling algorithm which favors advantaged users, and it is known that opportunistic scheduling algorithms achieve the highest average throughput (but at the expense of the fairness). Also, it is evident from Figure 2 that PF with OFDMA has higher throughput than that of PF with OFDM, as the former efficiently utilizes the resources in the frequency domain, and can handle efficiently the bursty traffic because of the subband sharing.

The Jain's fairness index of PF with OFDMA and PF with OFDM is depicted in Figure 3. Both algorithms show approximately the same values of Jain's fairness index with a slight improvement for PF with OFDMA. Also, we can notice that as T_c, increases (when $T_c > 0$), the fairness index decreases, as the algorithm becomes less fair (as discussed above). Furthermore, the lowest Jain's fairness index is associated with $T_c = 0$ because this is the case when PF becomes completely opportunistic, as discussed above.

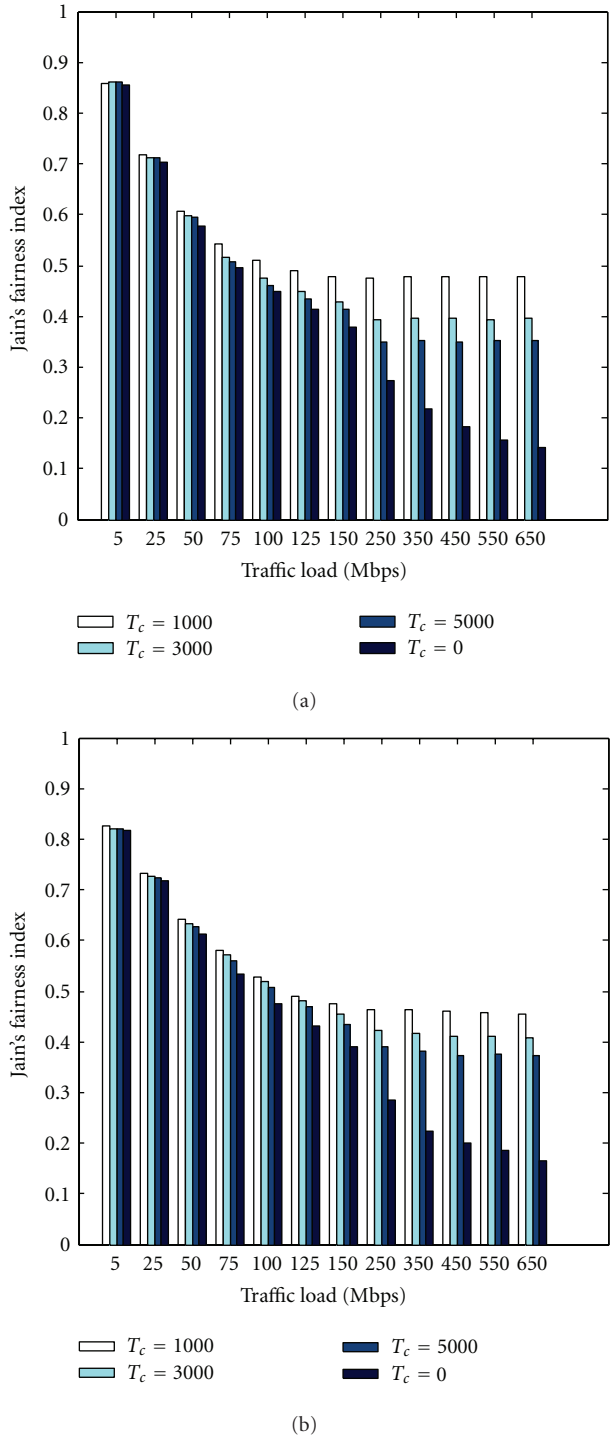


Figure 3: Fairness index of the PF scheduling with (a) OFDMA (b) OFDM.

In Figures 4 and 5, the throughput and the Jain's fairness index of the system are, respectively, shown versus the total traffic load in the cell. Results obtained from both analytical expressions in (20) and (21) and simulations are presented. It is noteworthy the good agreement between these results, which validate our analytical solution. From

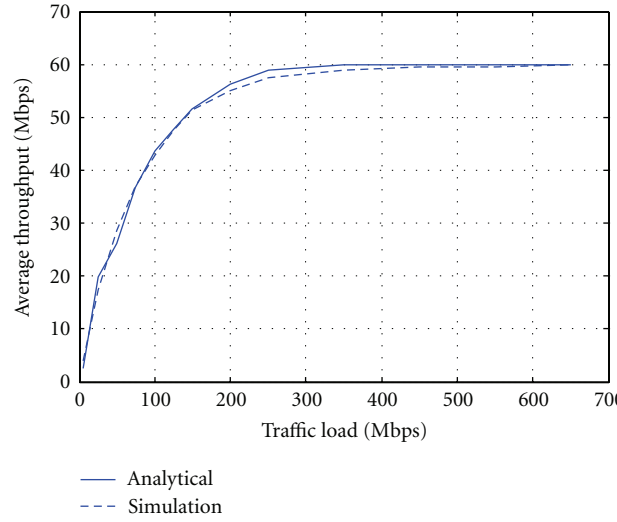


Figure 4: Average throughput versus traffic load.

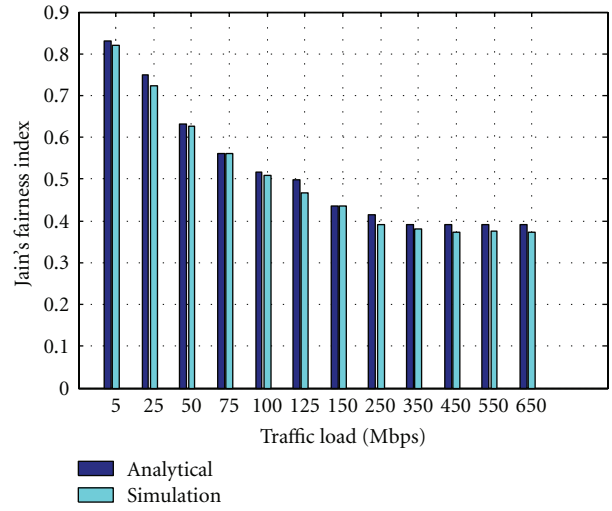


Figure 5: Jain's fairness index versus traffic load.

Figure 4, one can observe that (as expected) the average throughput increases sharply at low traffic load, and then it saturates at high traffic load. On the other hand, as shown in Figure 5, the fairness index decreases with the traffic load increase, and it saturates at high traffic load. This is because as the traffic load increases, fewer resources become available and it becomes more difficult to satisfy the demand of all users.

The performance of the PF scheduling algorithm that we propose in [8] and the agreement between analytical and simulation results are also investigated for a different number of users, N, where the traffic load expected from each user is assumed to be 10 Mbps and the averaging window, T_c, for the simulation, is selected to be 5000. Figures 6 and 7 show the average throughput and Jain's fairness index versus the number of users, respectively. Again, it is straightforward to notice that there is good matching between analytical and simulation results. From Figure 6, one can see the increase in

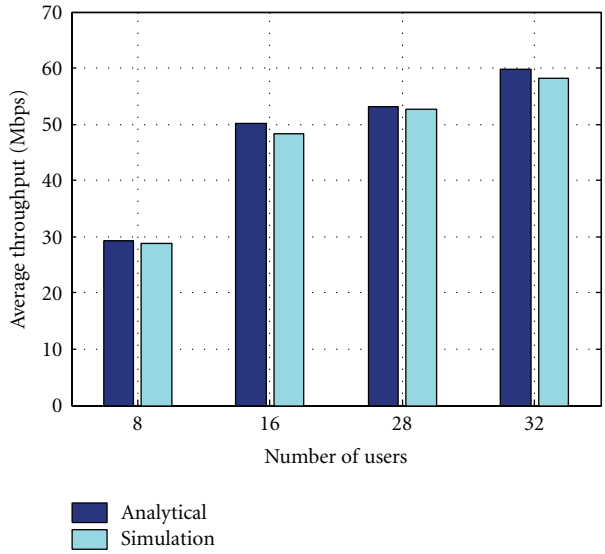


Figure 6: Average throughput versus number of users.

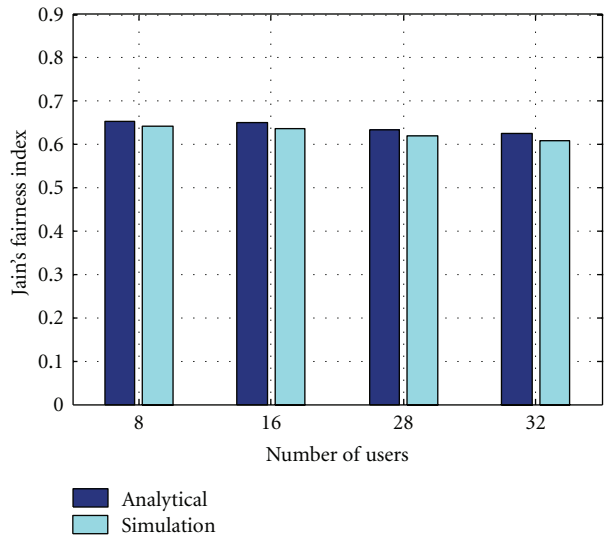


Figure 7: Jain's fairness index versus number of users.

the average throughput when the number of users increases for both analytical and simulation bars. This can be easily explained as follows: as the number of users increases, the traffic loads increase in the system. Also, as the number of users increases, the chance of scheduling users on subbands with preferable channel gain increases, so the scheduling algorithm utilizes the multiuser diversity. From Figure 7, we notice a slight fairness index decrease when the number of users increases. This fairness index decrease is expected, as the competition when the number of users increases.

Figure 8 shows the throughput performance at different number of subbands (M). The available frequency bandwidth is divided into different number of subbands to study the behavior of the system with different numbers of subbands. It is evident that the analytical results and the simulation results agree very well. We also notice that

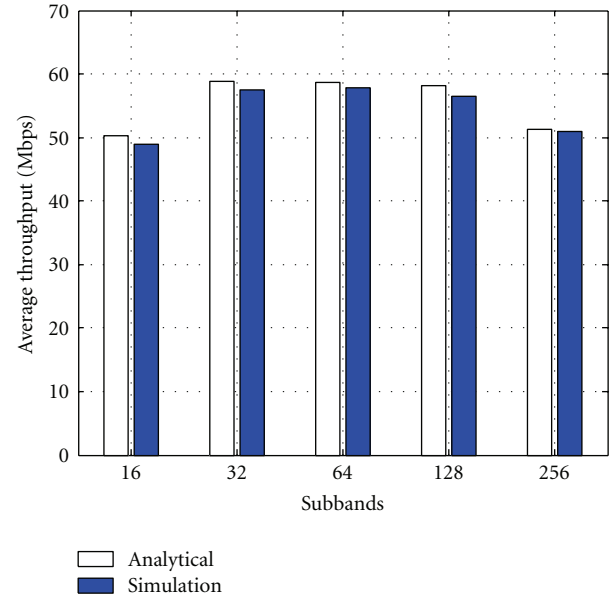


Figure 8: Throughput versus the number of subbands (traffic load = 250 Mbps).

the throughput reaches the maximum when the number of subbands equals 64. When the number of subbands is small, the number of subcarriers per subband is larger. Hence, the use of the adaptive coding and modulation for all the subcarriers, based on the subcarriers with worst channel conditions, will waste the resources of many subcarriers with favorable channel conditions. On the other hand, when the number of subbands is large, few subcarriers are grouped to create a subband, which degrades the throughput because of the increasing amount of unused fractions of subbands at the end of time frames. In other words, when the number of subbands increases, the number of subbands that are not fully utilized at the end of time frames increases, which degrades the throughput performance.

Figure 9 shows the Jain's fairness index at different number of subbands. We notice that the number of subbands does not affect the fairness of the system, as all users suffer from the same degradation of subbands utilization. Thus, the chance of accessing the resources will be affected equally for all users in the system, which keeps the fairness performance the same, regardless of the number of subbands.

Figure 10 shows the packet delay versus traffic load for the proposed scheduling algorithm, for T_c equals 5000, 3000, and 1000. It is evident that as the traffic load increases, the competition between users becomes harder, which causes more packets to wait longer time in the users queues. Also, we notice that when T_c increases, the packet delay increases. This can be explained as follows. When T_c increases, the scheduler tries to maximize the system throughput by forcing greedy treatment among users by allocating most of the resources to a few of users who have favorable channel conditions. That behavior blocks more packets for requesting users, which increases the average packet delay in the system.

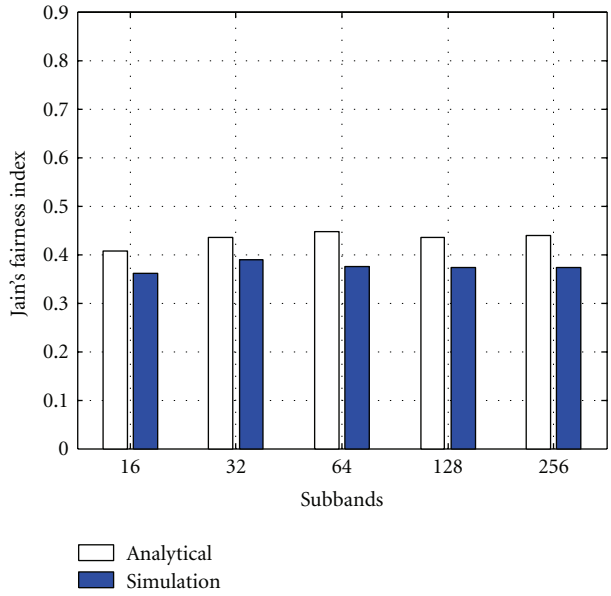


Figure 9: Jain's fairness index versus the number of subbands (traffic load = 250 Mbps).

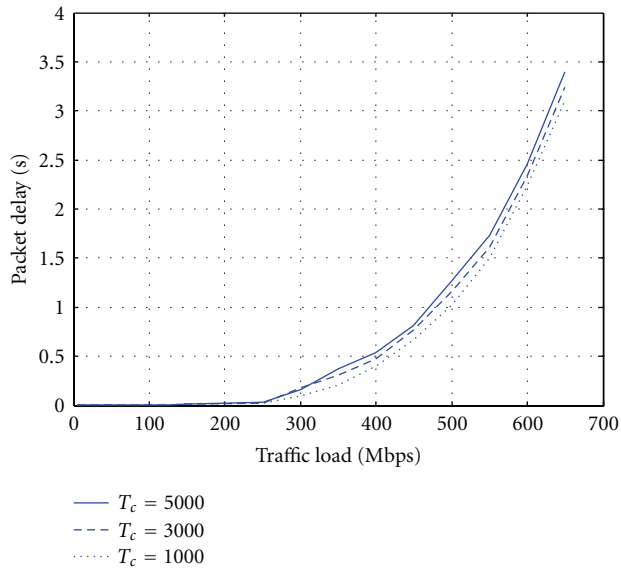


Figure 10: Mean packet delay versus traffic load of the proposed algorithm.

Figure 11 shows the packet delay versus traffic load for the proposed scheduling algorithm (PF with OFDMA), analytically and by simulation, and the packet delay for the PF with OFDM, where the observation window T_c equals 5000. As we notice, the analytical curve agrees very well with the simulation curve. Also, we notice a slight improvement of the proposed scheduling algorithm over the PF with OFDM. We notice that on high traffic load (650 Mbps) our proposed scheduling algorithm mean packet delay equals 3.75 seconds while the mean packet delay of PF with OFDM equals 3.45 seconds.

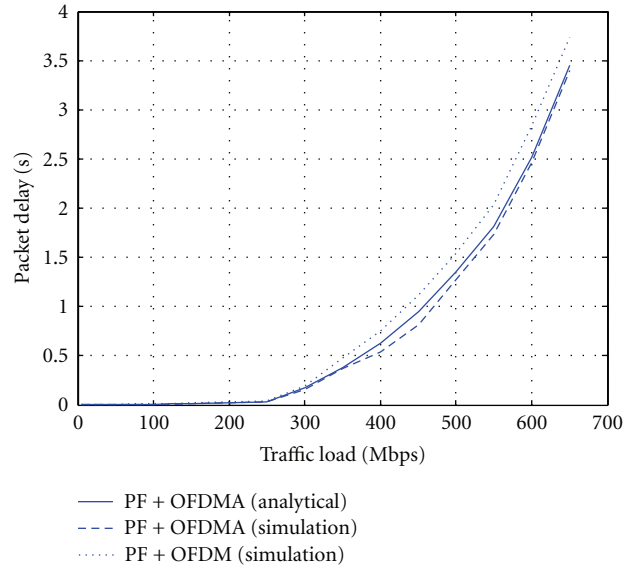


Figure 11: Mean packet delay versus traffic load for the proposed algorithm analytically and by simulation for PF with OFDMA and PF with OFDM.

It is noteworthy that there is a small difference between the analytical and simulation results. This result difference can be explained because of the approximations that have been introduced while deriving the analytical model. Such approximations simplify the model at the cost of minor result deviations.

6. Conclusion

In this work, the PF scheduling is investigated for OFDMA wireless systems. The main contribution of this work is the analytical evaluation of the performance of PF scheduling algorithm in OFDMA systems. We derive approximate closed-form expressions for the average throughput, Jain's fairness index, and packet delay as the performance metrics. The algorithm performance is investigated for a broad range of the traffic load and number of subbands. We compare the performance of the proposed algorithm (PF with OFDMA) with that of PF with OFDM. In addition, we verify the correctness and accuracy of the analytical solution through simulations. Analytical and simulation results are in good agreement, which validates our analytical performance analysis. In future work, we plan to extend the analysis to the case of different probabilities of the non-empty buffer for different users. We will also consider other fading distributions, such as the Rician distribution.

Acknowledgments

The authors are grateful to the anonymous reviewers and the editor for their constructive comments that improved the quality of the paper. This work has been supported by the NSERC Discovery Grant Program.

References

[1] L. C. Wang and W. J. Lin, "Throughput and fairness enhancement for OFDMA broadband wireless access systems using the maximum C/I scheduling," in *Proceedings of the IEEE 60th Vehicular Technology Conference (VTC '04)*, pp. 4696–4700, September 2004.

[2] Y. Ben-Shimol, I. Kitroser, and Y. Dinitz, "Two-dimensional mapping for wireless OFDMA systems," *IEEE Transactions on Broadcasting*, vol. 52, no. 3, pp. 388–396, 2006.

[3] C. Y. Wong, R. S. Cheng, K. B. Letaief, and R. D. Murch, "Multiuser OFDM with adaptive subcarrier, bit, and power allocation," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 10, pp. 1747–1758, 1999.

[4] I. C. Wong and B. L. Evans, "Optimal resource allocation in the OFDMA downlink with imperfect channel knowledge," *IEEE Transactions on Communications*, vol. 57, no. 1, pp. 232–241, 2009.

[5] H. J. Zhu and R. H. M. Hafez, "Scheduling schemes for multimedia service in wireless OFDM systems," *IEEE Wireless Communications*, vol. 14, no. 5, pp. 99–105, 2007.

[6] N. Ruangchaijatupon and Y. Ji, "Simple proportional fairness scheduling for OFDMA frame-based wireless systems," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 1593–1597, usa, April 2008.

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

[8] R. Almatarneh, M. Ahmed, and O. Dobre, "Frequency-time scheduling algorithm for OFDMA systems," in *Proceedings of the Canadian Conference on Electrical and Computer Engineering (CCECE '09)*, pp. 766–771, May 2009.

[9] E. Liu and K. K. Leung, "Proportional fair scheduling: analytical insight under Rayleigh fading environment," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 1883–1888, April 2008.

[10] B. Sklar, "Rayleigh fading channels in mobile digital communication systems Part I: characterization," *IEEE Communications Magazine*, vol. 35, no. 7, pp. 90–100, 1997.

[11] B. Sklar, "Rayleigh fading channels in mobile digital communication systems Part II: mitigation," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 148–155, 1997.

[12] L. C. Tran, T. A. Wysocki, A. Mertins, and J. Seberry, "A generalized algorithm for the generation of correlated Rayleigh fading envelopes in wireless channels," *Eurasip Journal on Wireless Communications and Networking*, vol. 2005, no. 5, pp. 801–815, 2005.

[13] P. J. Smith and M. Shafi, "On a Gaussian approximation to the capacity of wireless MIMO systems," in *Proceedings of the International Conference on Communications (ICC '02)*, pp. 406–410, May 2002.

[14] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," DEC Report DEC-TR-301, Digital Equipment Corporation, Littleton, Mass, USA, 1984.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 562615, 12 pages
doi:10.1155/2012/562615

Research Article

Optimizing Spectrum Trading in Cognitive Mesh Network Using Machine Learning

Ayoub Alsarhan[1] and Anjali Agarwal[2]

[1] *Department of Computer Information System, Hashemite University, Zarqa 13115, Jordan*
[2] *Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada H3G2W1*

Correspondence should be addressed to Ayoub Alsarhan, ayoubm@hu.edu.jo

Received 18 February 2012; Revised 6 May 2012; Accepted 16 May 2012

Academic Editor: Shuo Guo

In a cognitive wireless mesh network, licensed users (primary users, PUs) may rent surplus spectrum to unlicensed users (secondary users, SUs) for getting some revenue. For such spectrum sharing paradigm, maximizing the revenue is the key objective of the PUs while that of the SUs is to meet their requirements. These complex contradicting objectives are embedded in our reinforcement learning (RL) model that is developed and implemented as shown in this paper. The objective function is defined as the net revenue gained by PUs from renting some of their spectrum. RL is used to extract the optimal control policy that maximizes the PUs' profit continuously over time. The extracted policy is used by PUs to manage renting the spectrum to SUs and it helps PUs to adapt to the changing network conditions. Performance evaluation of the proposed spectrum trading approach shows that it is able to find the optimal size and price of spectrum for each primary user under different conditions. Moreover, the approach constitutes a framework for studying, synthesizing and optimizing other schemes. Another contribution is proposing a new distributed algorithm to manage spectrum sharing among PUs. In our scheme, PUs exchange channels dynamically based on the availability of neighbor's idle channels. In our cooperative scheme, the objective of spectrum sharing is to maximize the total revenue and utilize spectrum efficiently. Compared to the poverty-line heuristic that does not consider the availability of unused spectrum, our scheme has the advantage of utilizing spectrum efficiently.

1. Introduction

With the explosion of the number of emerging wireless applications for mobile users, the frequency spectrum has become congested to support the dramatic increase in the demand for the limited spectrum. Moreover, traditional spectrum management policies have contributed significantly in spectrum scarcity crisis [1]. In such schemes, the licensed spectrum is used only by the owner of license; other users are prevented from utilizing the unused spectrum. Consequently, spectrum owners are prevented from real-time interaction with radio environment and from determining appropriate communication parameters and adapting to the changes in the radio environment. For example, to increase data transfer rate and avoid interference, the wireless system may detect and switch to another lightly crowded band.

Fixed spectrum assignment policies prevent users from dynamically utilizing unused allocated spectrum; hence poor utilization and spectrum holes will be resulted. Moreover, the owner loses the profits from renting the unused spectrum. In wireless technology, another challenge is guaranteeing the QoS of applications that require huge bandwidth resources and service continuity protection [1]. To overcome the spectrum scarcity problem, the Federal Communications Commission (FCC) allows SUs to use unutilized spectrum if they do not interfere with PUs [1–3]. Recently, wireless mesh networks (WMNs) have emerged as a significant new technology that can provide ease of installation, low-cost means for flexible, and fast deployment of Internet-based services in diverse environments [1–3]. In order to become a mature technology, WMNs need to offer multimedia and emergency services that require more bandwidth resources.

Dynamic spectrum access (DSA) is proposed to mitigate spectrum scarcity through utilizing spectrum efficiently. It also enables users to adjust communication parameters (such as operating frequency, transmission power, and modulation scheme) in response to the changes in the radio environment [1–3]. DSA enables implementation of cognitive radio (CR) that brings a promise to increase spectrum at a minimum cost by using licensed spectrum whenever spectrum owners do not use it. This approach provides up to 85% of the unused spectrum [1]. CR also enhances the capability of WMNs to support broadband systems. CR encourages implementing new more flexible spectrum sharing paradigms. These sharing paradigms include the use of trading spectrum access on secondary market where PUs can rent unused spectrum to SUs and generate more revenue [3]. Despite of obvious advantages of using CR in WMNs, there are still several issues that require more investigation such as economic factors that include PUs revenues and SUs satisfaction. Spectrum trading also presents the challenge of sharing spectrum among primary users. This paper addresses when and how spectrum is shared among PUs and between primary and secondary users. Spectrum is shared between PUs and SUs based on our economic model and under dynamic traffic load conditions. Our economic model includes the costs and revenues associated with renting a spectrum. The cost of renting spectrum is a reduction of spectrum for PUs in favor of increasing revenue.

In our work, PUs borrow channels from other PUs. Our design objective is to improve spectrum utilization (among PUs) and maximize revenue for spectrum owners (spectrum trading), while meeting some defined constraints. In order to develop an intelligent radio that is able to deal with conflicting objectives in radio environment, we propose to use reinforcement learning which is an effective tool to deal with rational entities that make decisions to maximize their benefits with whatever little information they have [4]. It provides a mathematical framework for modeling decision-making in situations where the decision maker is not sure about the outcome.

In our work, reinforcement learning (RL) is used as a means for extracting an optimal policy that helps a PU to adapt to the changing radio environment conditions. PUs employ the extracted optimal policy to solve the following dilemma. When a request for spectrum arrives, the PU recognizes that it should give part of its spectrum to gain the revenue from rent. However, the QoS for PU might be degraded due to renting the spectrum. The PU might reject serving because it needs this spectrum and loses the reward. As a result, the PU waits for its demand for the spectrum to subside before renting spectrum. Consequently, the likelihood of losing a reward of serving SUs increases, which pushes the PU to become more spectrum-demanded in order to reduce its loss. Under the emerging secondary market spectrum policy, when renting available spectrum to other parties (i.e., PU, SU), the PUs need to consider the economic factors, such as the spectrum price and the operation revenue obtained. We formulate this spectrum trading problem as a revenue maximization problem. Such a formulation allows RL to optimize the trading problem. The contributions of our paper are as follows.

(i) A new spectrum-sharing scheme among PUs is proposed.

(ii) How the concept of RL can be used to obtain a computationally feasible solution to the considered spectrum trading problem is described.

(iii) An extensive numerical evaluation, based on analysis and simulation, of the RL-based method for spectrum trading is presented.

The rest of this paper is organized as follows. Firstly, we present previous work in spectrum sharing and trading, followed by our assumptions and work environment. We then describe our spectrum sharing scheme and formulate the spectrum trading problem. In the next section, we describe our model for solving the problem using RL, and illustrate its implementation and how we optimize obtained revenues using the RL algorithm. Next, we present some of the tests performed and show the behavior of the implemented system under different conditions. Finally, our last section concludes the paper.

2. Related Work of Spectrum Trading Using CRs

In a cognitive network, PUs can rent their unused spectrum to SUs. The problem of spectrum trading was considered in [5] where each node charges other nodes for relaying its traffic. The objective function is defined as the revenue obtained from transmitting the node traffic plus other nodes charges minus the price paid for other nodes along the route to the destination. In [6], multiple PUs sell unused spectrum resources to SUs to get monetary gains while SUs try to get permissions from PUs for accessing the rented spectrum. In order to maximize the payoffs of both primary and secondary users, game theory is used to coordinate the spectrum allocation among primary and secondary users through a trading process. The payoff of a PU is defined as the difference between the price of the sold spectrum and the cost of buying spectrum. However, the model does not consider the QoS of PUs.

In the framework proposed in [7], a PU may lease the owned spectrum to SUs in exchange for cooperation in the form of distributed space-time coding. For the PU, the main concern is maximizing its quality of service in terms of either rate or probability of outage, accounting for the possible contribution from cooperation. However, SUs compete among themselves for transmission within the leased time slot following a distributed power control mechanism. PU charges SUs for the leased spectrum in [8]. The problem is formulated as an oligopoly market competition and a noncooperative game is used to obtain the spectrum allocation for SUs. Nash equilibrium is considered as the solution of this game. In [9], it is extended to multiple PUs selling the spectrum to SUs. The model considers the behavior of other PUs to specify the price of spectrum. In [10], the advantages of employing market forces to address the issues of wireless spectrum congestion and the allocation

of spectrum are addressed. It is shown that when unlicensed spectrum is assigned to all competing SUs during periods of excess demand an inefficient outcome is likely to result. PUs compete to sell a spectrum to a set of buyers in [11]. Game theoretic approach is proposed to obtain the selling quantities and bidding price.

Several studies tackle the issue of spectrum sharing among PUs. In [12], PUs compete with each other to get the spectrum. To analyze the dynamic spectrum allocation of the unused spectrum bands to PUs, an auction theory was used. The problem was formulated as a multiunit sealed-bid sequential and concurrent auction. In [13], PUs dynamically compete for portions of available spectrum. PUs are charged by the spectrum policy server for the amount of bandwidth they use in their services. The competition problem is formulated as a non-cooperative game and new iterative bidding scheme that achieves Nash equilibrium of the operator game is proposed. In the proposed system in [14], two spectrum brokers offer a spectrum for a group of PUs. The broker wants to maximize its own revenue. Brokers' revenues are modeled as the payoffs that they gain from the game. On the other hand, PUs want to maximize its own QoS satisfaction at minimum expense.

Centralized regional spectrum broker distributes a spectrum among PUs in [15]. PUs do not own any spectrum; instead they obtain time bound rights from a regional spectrum broker to part of the spectrum and configure it to offer the network service. In [16], users adjust their spectrum usage based on a defined threshold called poverty line. A PU can borrow from its neighbors if the neighbors have number of idle channels greater than a poverty line. However, this scheme does not consider the availability of channels and the load of PU. It is possible that the neighbors have a number of idle channels less than their poverty line and these channels will be unused. Moreover, none of these schemes consider what follows.

(i) Utilizing spectrum efficiently: spectrum owners compete for spectrum to maximize their revenues regardless of efficient spectrum utilization.

(ii) Maximizing total revenues of PUs through utilizing the whole spectrum: the cooperation between PUs to maximize total revenues is neglected in these schemes.

(iii) Learning PUs a control policy to adapt the offered size of spectrum and spectrum price based on the changes in the radio environment such as traffic load, cost of services, and spectrum price.

Using simulations, we show the ability of our scheme to utilize spectrum efficiently by comparing its performance with the poverty-line scheme. Moreover, we conduct some experiments to show how our scheme can adapt to different network conditions such as traffic load and spectrum cost.

3. Network Overview

In this section, we present our assumptions. The network consists of two types of nodes: mesh routers (MRs) and mesh clients (MCs). A wireless mesh network has several MRs that jointly form a cluster [17]. Each cluster is a WLAN, where MRs play the role of access point and the MCs act as nodes served by them. The algorithm proposed in [17] is used to form and maintain clusters. Moreover, the proposed signaling protocol in [17] is used to manage communication among the PUs and the SUs. MRs have fixed locations, whereas MCs are moving and changing their places arbitrarily. The spectrum is divided into non-overlapping channels which are the basic unit of allocation. The network consists of W PUs and N SUs. We define a PU as a spectrum owner that may rent a spectrum to other users. Each PU has K channels assigned to it in advance. Each PU offers an adaptable number of channels to MRs (SUs). MRs use the rented spectrum to serve MCs. We assume that spectrum-request arrival follows Poisson distribution with arrival rate λ (the mean number of requests arriving per unit time). The service rate for incoming request is assumed to be exponentially distributed with service rate μ. These assumptions capture some reality of wireless applications such as phone call traffic.

4. Spectrum Trading Model

In this section, we formulate a theoretical model that is used to describe the general spectrum trading problem between PUs and SUs. Next we describe our on-demand-based spectrum sharing scheme and define the constraints of borrowing a spectrum among PUs.

4.1. Spectrum Trading Problem Formulation. In our model, we define the components for primary user $y(\mathrm{PU}_y)$ as follows.

(i) Spectrum allocation vector SP_y:

$\mathrm{SP}_y = \{\mathrm{SP}_y(m) \mid \mathrm{SP}_y(m) \in \{0,1\}\}$ is a vector of spectrum status. If $\mathrm{SP}_y(m) = 1$, channel m is not available currently. Spectrum status changes over time according to the spectrum demand.

(ii) Interference vector I_y:

$I_y = \{I_y(i) \mid I_y(i) \in \{0,1\}\}$ is a vector that represents the interference among PU_y and other PUs; if $I_y(i) = 1$, then PU_y and PU_i cannot use the same channel at the same time because they would interfere with each other.

(iii) Channel reward vector R_y:

$R_y = \{R_y(m) \mid R_y(m) \in \{0,\infty\}\}$is a channel reward vector, which describes the reward that PU_y gets by successfully renting channels to SUs. $R_y(m)$ is the reward that PU_y gets from renting channel m. It is computed as follows:

$$R_y(m) = p\mu\omega_m, \tag{1}$$

where p is spectrum price for renting a channel m and ω_m represents the quality of wireless transmission for channel m and is computed as follows:

$$\omega_m = \frac{\mathcal{C}\{m\}}{\arg\max_z \mathcal{C}\{z\}}, \tag{2}$$

where $\mathcal{C}\{m\}$ is the capacity of channel m and is computed using Shannon's formula. To fit the reward function in (1), channel m's capacity is normalized by the largest capacity among all channels. It is clear from (2) the channel with higher capacity provides high-quality communication and it should get higher reward than others. The average reward for a PU is computed mathematically as follows:

$$\overline{R} = r\overline{\lambda}, \tag{3}$$

where r is the reward of serving one request, and $\overline{\lambda}$ is the average rate of accepting SUs, defined as

$$\overline{\lambda} = \frac{A_c}{T_r}, \tag{4}$$

where A_c is the number of accepted requests, and T_r is the total number of requests. Equation (3) is used to compute analytical reward for a PU. The total reward TR_y is the following:

$$TR_y = \mathrm{SP}_y \cdot R_y^t, \tag{5}$$

where R_y^t is transpose of the channel reward vector.

(iv) Borrowable channel set BC_y:

our scheme allows two neighbors to exchange channels to maximize their reward while complying with conflict constraint from set of the neighbors. We define that two PUs are neighbors if their transmission coverage area is overlapped with each other. The set of channels that PU_y can borrow from PU_j should not interfere with PU_y neighbors. We refer to these channels as $\mathrm{BC}_y(\mathrm{PU}_y, \mathrm{PU}_j)$:

$$\mathrm{BC}_y\left(\mathrm{PU}_y, \mathrm{PU}_j\right) = \frac{L\left(\mathrm{PU}_j\right)}{L\left(G\left(\mathrm{PU}_y\right) \setminus \mathrm{PU}_j\right)}, \tag{6}$$

where L gives the set of channels assigned to the given user(s) (e.g., $L(\mathrm{PU}_j)$ represents the list of PU_j channels); $G(\mathrm{PU}_y)$ is a list of neighbors of a primary user PU_y.

4.2. On-Demand-Based Spectrum Sharing Scheme. In our scheme, PUs can exchange channels if the borrowed channels do not interfere with the channels of neighbors. After serving a request, the PU returns back borrowed channels to the owner users. PUs adjust their spectrum usage based on demand. As a result, the PU decides to borrow channels if the spectrum is not available to accommodate SUs requests and it is profitable to serve new SUs in terms of revenue. In our scheme, spectrum is shared among PUs as follows.

Step 1. PU computes the revenue of serving new SUs.

Step 2. If the revenue is positive and worthy, a PU requests neighboring PUs for a spectrum through a "borrowing frame" that is broadcast to all neighbors. The request frame specifies the size of required spectrum.

Step 3. Each PU receives a "borrowing frame," checks its idle channel list, and if there are idle channels, the PU temporarily gives up a certain amount of idle spectrum and sends an "accept frame" that includes channel IDs. If all channels are busy then the request is ignored.

Step 4. After receiving "accept frame(s)," the PU specifies a borrowable channel set BC and ranks its elements based on their capacity. If the PU does not receive any "accept frame," it queues the requests.

Step 5. After selecting channels, the PU informs the owners of the selected channels.

Step 6. After the PU finishes serving SUs, it returns the borrowed channels.

Our scheme guarantees high utilization through using all system channels provided that the interference constraint is met.

5. Reinforcement Learning-Based Model

Reinforcement learning is a subarea of machine learning concerned with how a system administrator takes actions in different circumstances in a work environment to maximize long-term revenue [4]. Let $X = \{X_0, X_1, X_2, X_3, \ldots, X_t\}$ be the set of possible states an environment may be in, and let $A = \{a_0, a_1, a_2, \ldots, a_t\}$ be a set of actions a learning agent may take. In RL, a policy is any function: $\pi : X \rightarrow A$ that maps states to actions. Each policy gives a sequence of states when executed as follows: $X_0 \rightarrow X_1 \rightarrow X_2 \ldots$ where X_t represents the system state at time t and a_t is the action at time t. Given the state X_t, the learning agent interacts with the environment by choosing an action a_t, then the environment gives a reward r_t and the system transits to the new state X_{t+1} according to the transition probability $P_{X,X_{t+1}}$ and the process is repeated. The goal of agent is to find an optimal policy $\pi^*(X)$ which maximizes the total reward over time. In this section, we define RL model applicable to control the spectrum trading.

5.1. Basic Formulation of RL Model. For the basic formulation, we describe the elements that facilitate the definition of the RL. These elements are the events and states of the system. Each PU has one finite FIFO queue for SUs (MRs) requests. The PU uses extracted optimal control policy to decide whether it is worthy to increase the offered spectrum for a new request or queue it. The request is added to the tail of the queue if a spectrum is insufficient to accommodate it and a PU fails to borrow a spectrum from other PUs. The request is served if the PU has sufficient spectrum. However,

if a queue is full, the request is rejected. In our work, the agent is developed to be implemented at the PU level of WMN in a distributed manner. It provides the trading functionality for a single queue. Each agent uses its local information and makes a decision for the events occurring in the PU in which it is located.

In our model, we have an adaptable spectrum size, $f(X_t)$, according to the percentage of queue usage (traffic load) and the gained revenue. At time t, the state of the system X_t is the number of accepted requests. Accepted requests is served immediately if there is adequate spectrum or they might be placed in the queue. Let $\{X_t, t \geq 0\}$ denote a random variable which represents system states, X is the state space. At state X_t, spectrum size $f(X_t)$ is used to serve the queued requests with a service rate $f(X_t)\mu$. Transition from one state to another means a request arrival or the SU is served. All possible states are limited by the following constraints:

(i) $X_t \leq QS$, where QS is the maximum length of the queue.

(ii) $f(X_t) \leq KW$,

where K and W are defined in Section 3. From a state, the system cannot make a transition (arrive, depart) unless the constraints are met.

5.2. Spectrum Trading Agent and State Space. In our system, an event can occur in a PU (agent) when a new request for spectrum arrives or a SU releases its assigned spectrum. These events are modeled as stochastic variables with appropriate probability distribution. At any time the PU is in a particular configuration defined by the size of offered spectrum for trading, the price of spectrum, and the number of admitted SUs.

In our case, each time a request for spectrum arrives one of the following decisions must be made: accept arrival request or reject the request. Upon serving the request, a PU has to decide the optimal offered-spectrum size for renting. The action space is given by

$$A = \{a : a \in \{0, 1\}\}, \tag{7}$$

where $a = 0$ denotes request rejection, $a = 1$ indicates that the PU has accepted the request and it might be placed in the queue if the spectrum is insufficient to accommodate it.

5.3. Model Optimization. In our model, the value $f(X_t)$ indicates the optimal spectrum size offered for SUs at state X_t that maximizes the estimated mean value of revenue,

$$\overline{V}(\pi^*) = \overline{R} - C, \tag{8}$$

where $\overline{R}$ is the average reward given in (3) and C is the cost of renting spectrum to the SUs and is computed as follows:

$$C = f(X_t) * \delta, \tag{9}$$

where δ is the cost of renting one spectrum unit to SUs. Due to spectrum renting, the spectrum remaining for the primary user becomes smaller; hence its QoS is degraded. The rate of revenue at state X_t is computed as

$$q(X_t) = a_t(r * X_t * f(X_t)\mu - C), \tag{10}$$

where r is the reward for renting spectrum and is computed using (1). The actual mean value of the net revenue under policy π for PU_y is given by

$$\overline{A}_y(\pi) = \frac{\lim_{D\to\infty} \sum_{t=1}^{D} q(X_t)}{D}, \tag{11}$$

where $\overline{A}_y$ indicates the average actual value of the net revenue of PU_y when policy π is executed and D represents the time horizon. The state transition probability is given by

$$P_{X_t,X_{t+1}}(a) = \left\{ \begin{array}{ll} \lambda, & X_{t+1} = X_t + 1 \\ f(X_t)\mu, & X_{t+1} = X_t - 1 \\ 0, & \text{otherwise} \end{array} \right\}, \tag{12}$$

where X_t represents current state and X_{t+1} is the next transited state. In our system, an event can occur in a PU (agent) when a new request for spectrum arrives or a SU releases its assigned spectrum. These events are modeled as stochastic variables with appropriate probability distribution. Hence, the state transition occurs when a request arrives or is served and this is shown in (12).

5.4. Optimal Policy. The optimal policy gives the maximum net revenue when a PU adopts it. It specifies the optimal spectrum size and price for each state. Basically, in our model the optimal policy is specified according to the average revenue value obtained for each transition with the offered spectrum size. For each state, the revenue gained depends on the action reward, cost of spectrum, and the spectrum demand. When a new spectrum request arrives at the queue, the PU checks if it is worthy to increase the offered spectrum based on the revenue gained. It then either increases the offered spectrum or keeps it. When a SU departs from the system the PU may decrease the offered spectrum based on revenue. Although decreasing the size of spectrum decreases the customers' satisfaction—since their waiting time will increase accordingly—the PU always chooses the action that maximizes its revenue. In our work, a PU uses RL to choose a policy, $\pi : X \rightarrow A$, for deciding the next action a_t based on the current state X_t. We apply a value iteration algorithm to find an optimal policy. The value function [4] of policy π is given as

$$V^{\pi}(X_t) = q(X_t) + \alpha \sum_{X_{t+1}\in X} P_{X_t,X_{t+1}}(a) V^{\pi}(X_{t+1}), \tag{13}$$

where α is the discount revenue that satisfies $0 \leq \alpha < 1$ and starting with $t = 0$. The value function $V^{\pi}(X_t)$ can be considered as the expected revenue for policy π starting from state X_0. The optimal value function is given [4] as

$$V^*(X_t) = q(X_t) + \max_{a\in A} \alpha \sum_{X_{t+1}\in X} P_{X_t,X_{t+1}}(a) V^*(X_{t+1}). \tag{14}$$

The optimal policy is given as follows [4]:

$$\pi^*(X_t) = \arg\max_{a\in A} \sum_{X_{t+1}\in X} P_{X_t,X_{t+1}}(a)V^*(X_{t+1}). \quad (15)$$

We define an optimal policy π^* as follows:

$$H_y(\pi^*) \geq H_y(\pi), \quad (16)$$

where H_y indicates the total net revenue of PU_y computed as follows:

$$H_y(\pi^*) = \lim_{D\to\infty} \sum_{t=1}^{D} q(X_t). \quad (17)$$

5.5. Analytical Model for Spectrum Trading. Network conditions are changing randomly. These conditions include traffic level, spectrum cost, and the size of unused spectrum. As a consequence, PUs should adapt to continue increasing the revenue. The principal parameters that PUs control are the price and the size of the offered spectrum. In our model, the PUs' revenues sensitivity to the number of the offered spectrum size ($f(X_t)$) can be derived from (8):

$$\frac{\partial \overline{V}}{\partial f(X_t)} = \left(\frac{\partial \overline{R}}{\partial f(X_t)}\right) - \left(\frac{\partial C}{\partial f(X_t)}\right) = \left(\frac{\partial \overline{R}}{\partial f(X_t)}\right) - \delta. \quad (18)$$

We assume the average reward sensitivity to the spectrum size can be approximated by the cost of accepting new SUs, u, which is calculated as follows:

$$u = r - o, \quad (19)$$

where o is the reward increment from accepting new requests. Substituting in (18), the PU's revenue is maximized when spectrum size equals the root of

$$\frac{\partial \overline{V}}{\partial f(X_t)} = u(f(X_t)) - \left(\frac{\partial C}{\partial f(X_t)}\right) = 0. \quad (20)$$

We used Newton's method of successive linear approximations to find the root of (20). The new spectrum size $f(X_t)_{n+1}$ at each iteration step n is computed as follows:

$$f(X_t)_{n+1} = f(X_t)_n - \frac{u_n - \delta}{\partial(u(f(X_t)) - \delta)/\partial f(X_t)}. \quad (21)$$

Approximating the derivative in (21) at step n:

$$\frac{\partial(u(f(X_t)) - \delta)}{\partial f(X_t)} = \frac{\partial u(f(X_t))}{\partial f(X_t)} \cong \frac{u_n - u_{n-1}}{f(X_t)_n - f(X_t)_{n-1}} \quad (22)$$

and substituting (22) in (21), the new spectrum size will be

$$f(X_t)_{n+1} = f(X_t)_n - \left(f(X_t)_n - f(X_t)_{n-1}\right)\frac{u_n - \delta}{u_n - u_{n-1}}. \quad (23)$$

Spectrum size adaptation is then realized using Algorithm 1, where ε is the tolerable error. The presented solution for revenue maximization does not take into account the QoS of PUs. The request of spectrum from the PU is blocked if it arrives while a PU is already using all of its spectrum. Therefore, the probability of blocking for PU_y is computed as follows [18]:

$$B_y = \frac{\rho^K}{K!}\left(\sum_{k=0}^{K}\left(\frac{\rho^K}{K!}\right)^{-1}\right), \quad (24)$$

where ρ is computed as follows:

$$\rho = \frac{\lambda}{\mu}. \quad (25)$$

Although for optimal spectrum size and price, one can expect that standard blocking constraint of PUs will be met. However, in some scenarios the blocking probabilities may exceed the constraints. To cope with this constraint, we use a spectrum price for controlling the size of the offered spectrum and meeting the blocking probability for the PUs. It is clear when a PU increases the price of spectrum the arrival rate of SUs and the demand of spectrum will be decreased. The arrival rate depends on the offered price. The new arrival rate of SUs is calculated as follows [19]:

$$\lambda = \tau e^{-\varphi \acute{p}}, \quad (26)$$

where τ is the maximum number of users arriving to a PU, φ represents the rate of decrease of the arrival rate as spectrum price increases and is related to the degree of competition between the PUs, and $\acute{p}$ is the new price. Here, we assume φ is given a priori. There is an inverse relationship between the price and the demand of the spectrum. A PU has to meet its blocking probability constraint B_y^C. Blocking probability depends on the number of available channels and the traffic load. If the blocking probability for a PU exceeds the blocking constraint, a PU continues to increase the spectrum price till its blocking probability is met. Because of the inverse relationship between spectrum price and its demand, it can be easily shown that when a spectrum price is increased the available channel for a PU will be increased and therefore the blocking probability will be reduced. This feature indicates that if a PU_y blocking constraint B_y^C is not met, $B_y > B_y^C$, the spectrum price is increased to fulfill the blocking probability constraint. However, we assume that the potential price increment should be minimized as possible as it can keep the demand for spectrum-high and maximize the PUs revenues. After increasing the spectrum price, the new revenue is computed as follows:

$$\Delta \acute{V} = \lambda(\acute{p} - p) - C. \quad (27)$$

```
AdaptSpectrumSize (u_n, f(X_t)_n, f(X_t)_{n-1}, δ, ε)
begin
if ((Abs(u_n − δ) < ε))
return f(X_t)_n, u_n;
else
      {
compute new value of u_n and f(X_t)_{n+1};
AdaptSpectrumSize (u_n, f(X_t)_{n+1}, f(X_t)_n, δ, ε);
      }
end;
```

Algorithm 1

This leads to the following problem formulation:

$$\begin{aligned} &\max_{f(X_t)} \overline{V} = \lambda \acute{p} - C - \min_{\acute{p}} \lambda(\acute{p} - p) \\ &\text{subject to} \quad \sum_{y=1}^{W} \mathrm{SP}_y \le KW, \\ &\qquad SP_y(c)\mathrm{SP}_j(c)I_y(j) = 0, \\ &\qquad B_y \le B_y^C, \\ &\qquad \Delta V' = \lambda(\acute{p} - p) - C \ge 0. \end{aligned} \tag{28}$$

In our proposed adaptation scheme the new values of spectrum prices reflect the amount of spectrum required by a PU. Due to the competition in the market, a price increment is limited due to the possibility of losing customers. If a blocking constraint for a PU is met it tries to meet the blocking constraint for SUs by increasing the offered spectrum size using the *AdaptSpectrumSize* algorithm.

Table 1: Simulation parameters.

Parameter	Value
Number of mesh routers	10
Number of clients	100
Number of primary users	10
Number of channels per a PU	20
Total number of channels	200
Number of messages per client	Random
Type of interface per node	802.11 b
MAC layer	IEEE 802.11 b
Transmission power	0.1 watt
Packet size	512
λ	1
Channel bandwidth	100 kHz
Blocking probably constraint for a PU	0.015
SNR	4 db

6. Performance Evaluation

In this section, we show simulation results to demonstrate the ability of our spectrum scheme to adapt to different network conditions. The system of PUs and SUs is implemented as a discrete event simulation. The simulation is written by using Matlab. We uniformly distribute 10 PUs and each PU is randomly assigned 20 channels. For the mesh network, 100 MCs are distributed uniformly in the transmission region of the MRs. The results presented are for several system settings scenarios in order to show the effect of changing some of the control parameters. The network parameters chosen for evaluating the algorithm and the methodology of the simulation are shown in Table 1.

6.1. Impact of Spectrum Size and Number of Primary Users on Spectrum Borrowing among PUs. Simulations are done to explore the availability of channels that can be borrowed under different configurations of spectrum size and primary user deployment. We vary the number of PUs and the number of channels (spectrum size). We assume that two users interfere if the distance between them is less than 20 m and they use the same channel. Figure 1 shows the borrowing probability for different numbers of PUs. We calculate the probability of existing channels being available for borrowing. Simulations are done to investigate the effect of the number of PUs on the probability of channel borrowing. We can see that the possibility of adjusting spectrum based on borrowing is not guaranteed for a large number of PUs with a small size of spectrum. Moreover, it can be seen that increasing the number of PUs, the borrowing probability decreases due to the interference among users. The spectrum size is another factor that influences channel borrowing probability. Increasing the size of spectrum (i.e., increasing the number of channels in the system) reduces the likelihood of interference.

6.2. Performance of On-Demand Sharing Scheme. We compare the performance of our on-demand-based spectrum sharing scheme with the poverty-line heuristic [16] through simulations. For PU_y, the poverty line is computed as follows:

$$\mathrm{PL}(y) = \frac{L\left(\mathrm{PU}_y\right)}{\mathrm{NG}\left(\mathrm{PU}_y\right)}. \tag{29}$$

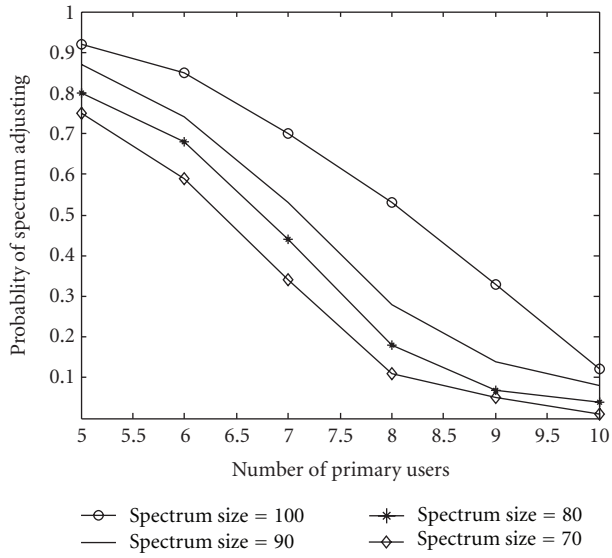


Figure 1: Probability of spectrum adjusting.

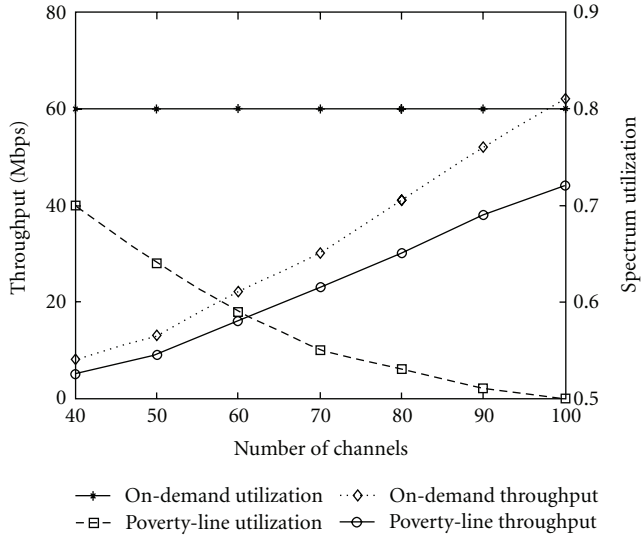


Figure 2: Throughput and spectrum utilization comparison.

The performance metrics considered are as follows.

(1) Throughput, which is the average rate of successful message delivery over a communication channel which can be expressed as follows:

$$\text{Throughput} = \frac{\text{total no. of bytes received}}{\text{simulation time}}. \quad (30)$$

(2) Spectrum utilization, S, which is the percentage of busy spectrum at time t and is computed as follows:

$$S = \frac{\sum_{w=1}^{W} \text{SP}_w}{KW}. \quad (31)$$

We examine the performance under different parameter settings. Throughput comparison of the two schemes is shown in Figure 2. The figure shows that the throughput increases as the total number of channels increases. This is due to more spectrum that can be employed. Our scheme utilizes the unused spectrum resourcefully because there is no limit to channel borrowing among PUs. For poverty-line heuristic [16], a PU cannot exceed a certain number of channels that can be borrowed from its neighbors even if the neighbors have idle channels.

We further present the results of spectrum utilization with different spectrum sizes in Figure 2. Our scheme performs better than the poverty-line heuristic. Our scheme utilizes the whole spectrum because PUs can have access to neighbor's channels based on availability of channels and on-demand. This improves the cognitive network throughput and overall spectrum utilization. However, some unused spectrum is not utilized under poverty-line heuristic because of the threshold constraint. It is clear from Figure 2 that our scheme is not sensitive to the number of channels in the network. However, the only constraint that prevents our scheme from full utilization of spectrum is the interference factor. In the poverty-line-based scheme, spectrum sharing is limited by the poverty line that depends on the number of idle channels. From the figure, we can see that as the number of channels increases the utilization of channels decreases because of an increment in idle channels.

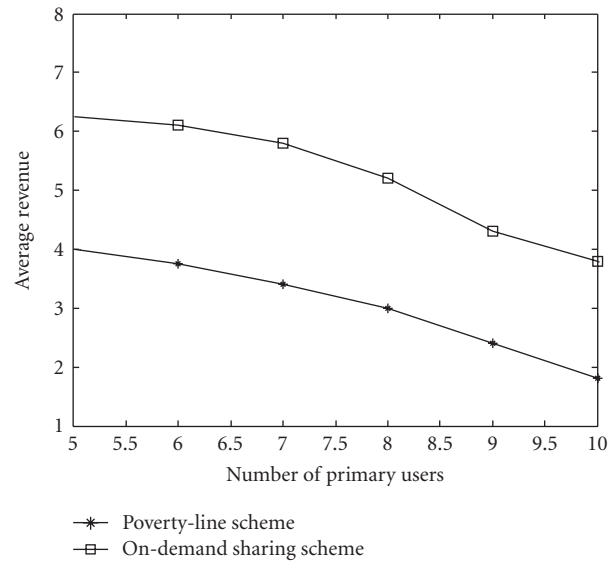


Figure 3: Average revenue sensitivity versus number of PUs in cognitive network.

Figure 3 displays the result of spectrum trading. The result shows that our scheme achieves higher revenue than poverty-line scheme. The revenue decreases as the number of PUs increases, since in this case PUs are assigned less number of channels; therefore the size of offered spectrum will decrease. We also compare the performance of the two schemes under varying traffic load in Figure 4. The result shows that the revenue increases as the spectrum demand increases.

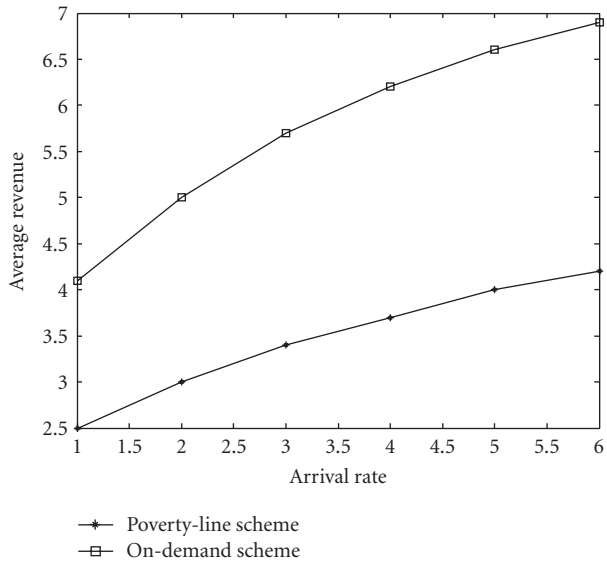


Figure 4: Average revenue sensitivity versus arrival rate.

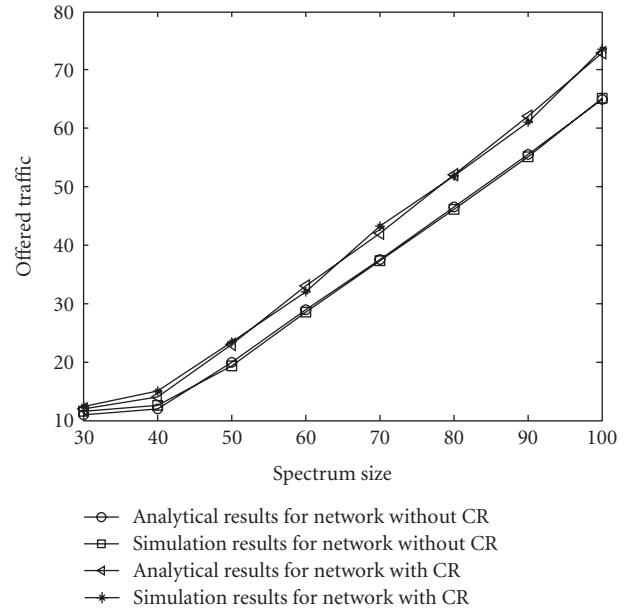


Figure 5: Offered traffic for WMNs with and without CR abilities (blocking probability).

6.3. Supporting QoS for SUs in CRs. In this section, we explore the performance of WMNs with cognitive abilities. CRs take advantage of surplus spectrum by renting it to the SUs and getting profits. Figure 5 shows a comparison between the traffic for WMNs with CR abilities and WMNs without CR abilities. Clearly, the cognitive systems outperform the classic WMNs that do not use CR technology. The main disadvantage of CRs is the waiting time of flows. This is a direct consequence of the PUs requirement of not renting a surplus spectrum if there is no revenue. However, despite the PUs requirement, the overall performance is far better when CR is enabled. CRs cannot guarantee QoS because PUs flows

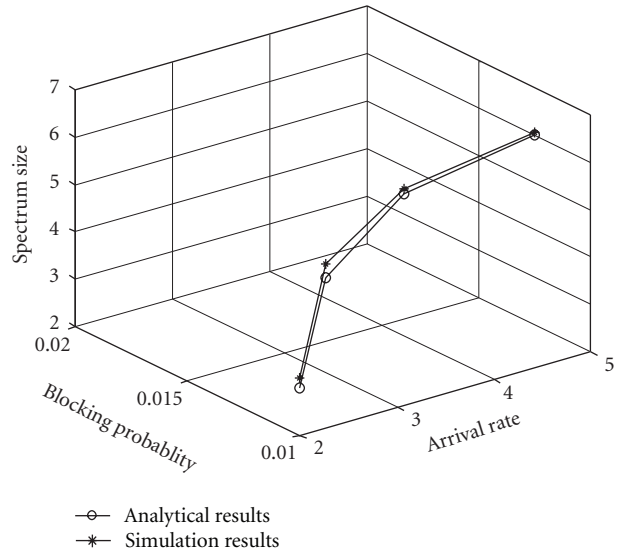


Figure 6: Adapting spectrum size to meet blocking probability constraint.

have a priority over SUs. Each PU needs a spectrum for its usage and to support the maximum classic traffic for ($B_y^C \leq 1\%$) constraint. If an additional network overlays its traffic over the unused spectrum it should not affect the B_y^C of the PUs.

6.4. Spectrum Price Adaptation. A PU with well-dimensioned spectrum size and correctly chosen spectrum price provides the desired QoS and maintains blocking probabilities in an acceptable range. When the spectrum demand increases, blocking probabilities normally increase beyond their constraints. While our adaptation scheme tries to maximize PUs' revenues by increasing spectrum size when the spectrum demand increases, it maintains QoS by bringing blocking probabilities back to its constrained range by increasing the spectrum price. Our intelligent algorithm is converged after 4 steps. Figure 6 displays the offered spectrum size at PU_y for different arrival rates. When spectrum arrival rate is increased and blocking probability does not surpass B_y^C, PU_y adapts by increasing the size of the offered spectrum as shown in the figure to generate more revenue. However, when the demand decreases, PU_y reduces the size of the offered spectrum to avoid a waste of spectrum.

We study the effect of spectrum adaptation on the gained revenue for different offered spectrum sizes in Figure 7. The results show that our algorithm increases the offered spectrum size to gain more revenue. When the offered spectrum becomes large the quality of service of PU may be degraded because of the reduction of its spectrum size. Therefore, the adaptation scheme stop increasing the offered spectrum. Figure 7 also shows the size of offered spectrum for different service costs. It is clear the adaptation scheme offers more spectrum when the cost of serving SUs is low. When a PU_y offers large size of spectrum, its blocking probability B_y may surpass its blocking constraint B_y^C. The spectrum price adaptation is integrated in our adaptation

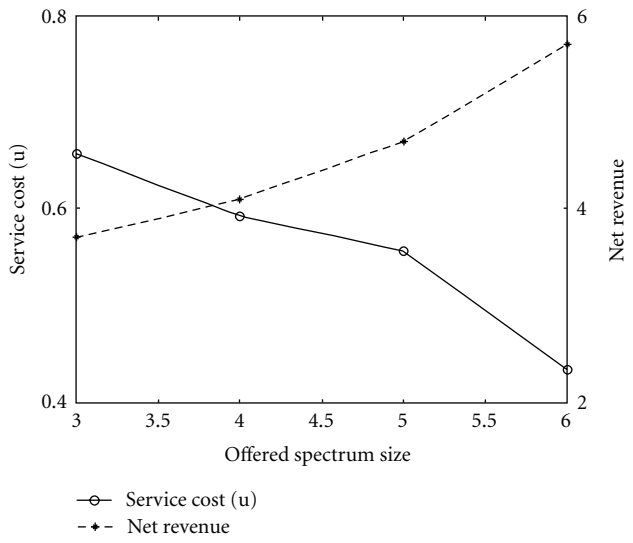


Figure 7: Adapting spectrum size for different service cost and the gained average revenue for the adaptation scheme.

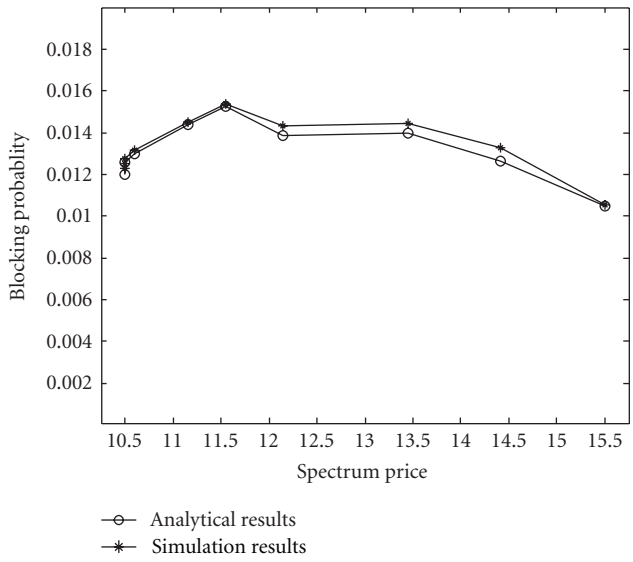


Figure 8: Adapting spectrum price to meet blocking probability constraint.

process to ensure it meets the blocking constraints. Figure 8 shows the spectrum price adaptation when the blocking probability surpasses blocking constraint. PU increases the price of spectrum to decrease the accepting rate for each SUs class and to maintain QoS for PUs. The results show our scheme's ability to bring blocking probabilities back to their constrained range by adapting spectrum price.

6.5. Tradeoffs between a PU Revenue and QoS Constraints. Figure 9 plots the tradeoff between a PU revenue and its QoS. To show the relationship between the two, we vary the blocking probability constraint for a PU (the QoS requirement for a PU). Blocking constraint becomes stricter in such a way that more in-service primary users should be

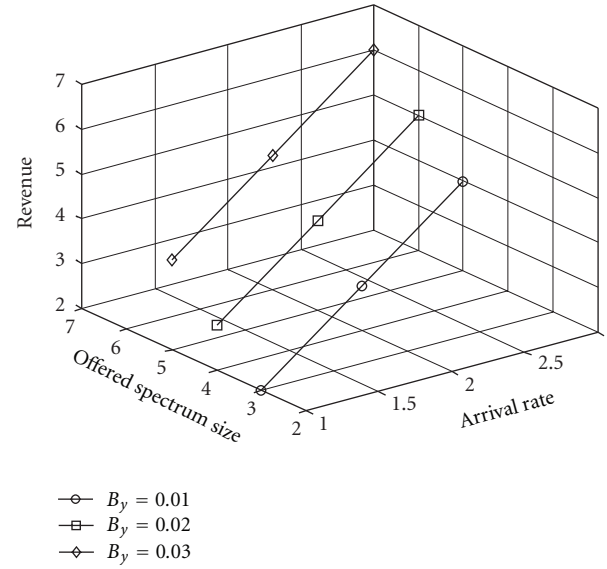


Figure 9: Offered spectrum for different blocking network constrains.

protected from channel eviction. For this, SUs arrivals must be blocked more often and the rejection ratio is increased. As a result, a PU cannot offer more spectrum for a small value of blocking probability. However, as this constraint is relaxed, a PU can serve more SUs and can offer more spectrum to generate more revenue. For large values of blocking probability, a PU can maintain a QoS for its applications and this can be observed from the figure. The revenue gained for large values of blocking probability is increased and a PU becomes less strict so that a lower number of SUs are rejected upon their arrival.

6.6. Optimal Policy as a Function of Spectrum Price, Cost, and Quality. We simulate the behavior of the described system under different spectrum prices. Figure 10 displays the size of the offered spectrum for different service prices. From the figure, we can clearly see that even though spectrum prices are higher a PU may increase the offered spectrum size. There is a direct correlation between the offered spectrum size and the spectrum price, so the more reward we have (due to price) the more spectrum PU can offer SUs. However, a PU cannot further increase the price because it will affect SU's spectrum demand.

We compare the same system for different cost of service (δ) in Figure 11 for a fixed spectrum price. From the figure, we clearly notice how sensitive the optimal size of offered spectrum is to the spectrum cost, where the offered size drops as spectrum cost increases. Figure 12 shows the offered spectrum size as a function of spectrum quality (ω). It is clear as the spectrum quality improves, the PU will offer more spectrum increasing its reward.

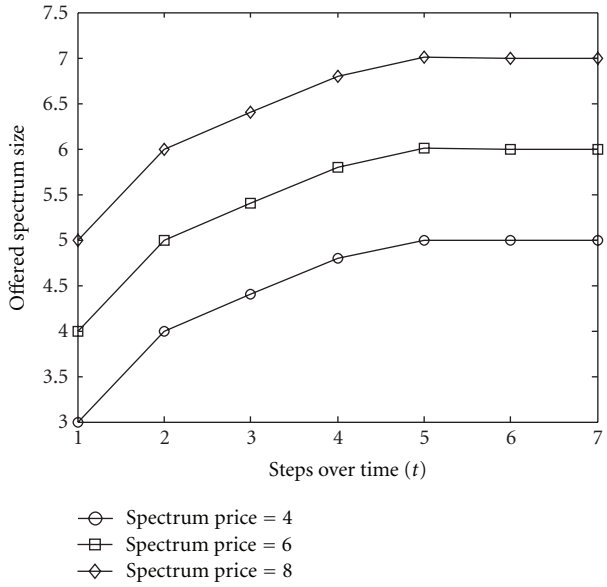


Figure 10: Optimal policy as a function of spectrum prices.

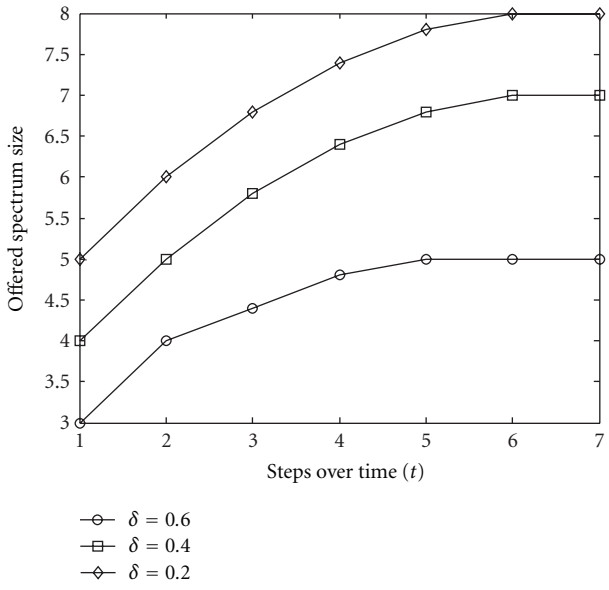


Figure 11: Offered spectrum size for different spectrum cost.

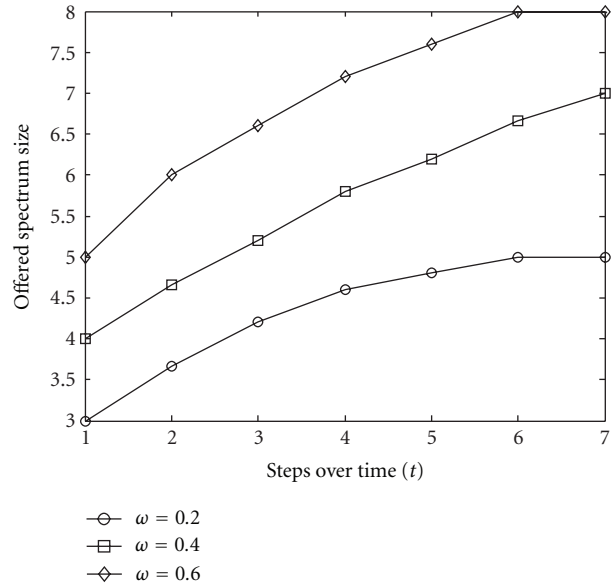


Figure 12: Offered spectrum size as a function of spectrum quality.

7. Conclusion

In this paper, we present a novel machine-learning-based model to obtain an optimal policy for controlling spectrum trading in cognitive wireless networks. The proposed model has two contributions to cognitive networks. From the application side, the main contribution is developing a control policy that considers different requirements such as rewards for PUs, wireless requirement (channel interference), the cost of spectrum renting, and PUs QoS. All basic functions are integrated and optimized into one homogenous, theoretically based model. From the modeling side, we formulate a spectrum trading problem as a reward maximization problem. Such a formulation allows RL to optimize the trading problem. The approach presents a general framework for studying, analyzing, and optimizing other resource management in cognitive mesh networks.

Another contribution is to propose a new scheme for the PUs to control spectrum trading for the emerging spectrum secondary market. PUs can employ the proposed scheme to choose the optimal price and size of the offered spectrum. The objective is to adapt the size and price of spectrum in order to continuously maximize PUs' net revenues while maintaining PUs' QoS. Simulations were also conducted and shown to closely agree with the analytical model. They demonstrated the ability of our algorithm to support SUs requirements and obtain the potential performance gains by applying cognitive radio. Moreover, the numerical results show that the proposed approach is able to find an efficient tradeoff between different rates of spectrum size and different costs of spectrum. The results show the ability of our scheme to find the optimal spectrum size for different spectrum prices. We vary system parameters to understand the behavior of the system under different scenarios. The results show a direct correlation between the reward rates and the spectrum price, and an inverse relationship between the spectrum cost and the allocated bandwidth. We also propose a new distributed spectrum sharing scheme among primary users. PUs share spectrum based on demand whereby they can borrow spectrum from their neighbors while complying with interference rules. The benchmark in our experiments is the poverty-line heuristic used in [16]. Because it can more efficiently employ limited spectrum resources compared to the poverty-line heuristic, our scheme achieves higher net revenues. The poverty-line heuristic restricts borrowing by a threshold called poverty line. Moreover, numerical results

show that our scheme is able to find an efficient tradeoff between PU revenues and SUs delay.

References

[1] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey," *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006.

[2] E. Hossain and V. K. Bhargava, *Cognitive Wireless Communication Networks*, Springer, New York, NY, USA, 2007.

[3] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 4, pp. 445–487, 2005.

[4] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, The MIT Press, Cambridge, Mass, USA, 1998.

[5] R. K. Lam, J. C. S. Lui, and C. Dah-Ming, "On the access pricing issues of wireless mesh networks," in *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS '06)*, pp. 61–61, 2006.

[6] Z. Ji and K. J. R. Liu, "Belief-assisted pricing for dynamic spectrum allocation in wireless networks with selfish users," in *Proceedings of the 3rd Annual IEEE Communications Society on Sensor and Ad hoc Communications and Networks (SECON '06)*, pp. 119–127, September 2006.

[7] O. Simeone, I. Stanojev, S. Savazzi, Y. Bar-Ness, U. Spagnolini, and R. Pickholtz, "Spectrum leasing to cooperating secondary ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 203–213, 2008.

[8] D. Niyato and E. Hossain, "Competitive spectrum sharing in cognitive radio networks: a dynamic game approach," *IEEE Transactions on Wireless Communications*, vol. 7, no. 7, pp. 2651–2660, 2008.

[9] D. Niyato and E. Hossain, "Market-equilibrium, competitive, and cooperative pricing for spectrum sharing in cognitive radio networks: analysis and comparison," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4273–4283, 2008.

[10] M. M. Bykowsky, M. Olson, and W. W. Sharkey, "Efficiency gains from using a market approach to spectrum management," *Information Economics and Policy*, vol. 22, no. 1, pp. 73–90, 2010.

[11] H. Sartono, Y. H. Chew, W. H. Chin, and C. Yuen, "Joint demand and supply auction pricing strategy in dynamic spectrum sharing," in *Proceedings of the IEEE 20th Personal, Indoor and Mobile Radio Communications Symposium (PIMRC '09)*, pp. 833–837, September 2009.

[12] S. Sengupta and M. Chatterjee, "Sequential and concurrent auction mechanisms for dynamic spectrum access," in *Proceedings of the 2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom '07)*, pp. 448–455, August 2007.

[13] O. Ileri, D. Samardzija, T. Sizer, and N. B. Mandayam, "Demand responsive pricing and competitive spectrum allocation via a spectrum server," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 194–202, November 2005.

[14] G. Işiklar and A. B. Bener, "Brokering and pricing architecture over cognitive radio wireless networks," in *Proceedings of the 5th IEEE Consumer Communications and Networking Conference (CCNC '08)*, pp. 1004–1008, January 2008.

[15] M. M. Buddhikot, P. Kolody, S. Miller, K. Ryan, and J. Evans, "DIMSUMNet: new directions inwireless networking using coordinated dynamic spectrum access," in *Proceedings of the IEEE WoWMoM*, pp. 78–85, 2005.

[16] L. Cao and H. Zheng, "Distributed rule-regulated spectrum sharing," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 130–145, 2008.

[17] A. Alsarhan and A. Agarwal, "Cluster-based spectrum management using cognitive radios in wireless mesh network," in *Proceedings of the 18th International Conference on Computer Communications and Networks (ICCCN '09)*, San Francisco, Calif, USA, August 2009.

[18] P. Beckmann, *Elementary Queuing Theory and Telephone Traffic*, Series on Telephone Traffic, Lee's ABC of the Telephone, Geneva, Switzerland, 1977.

[19] G. Gallego and G. van Ryzin, "Optimal dynamic pricing of inventories with stochastic demand over finite horizons," *Management Science*, vol. 40, no. 8, pp. 999–1020, 1994.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 781520, 11 pages
doi:10.1155/2012/781520

Research Article

Comparison Study of Resource Allocation Strategies for OFDM Multimedia Networks

Cédric Guéguen[1] and Sébastien Baey[2]

[1] *IRISA/Université de Rennes 1, Campus de Beaulieu, 35042 Rennes cedex, France*
[2] *LIP6/CNRS, UPMC Sorbonne Universités, 4 place Jussieu, 75005 Paris, France*

Correspondence should be addressed to Cédric Guéguen, cedric.gueguen@irisa.fr

Received 23 February 2012; Revised 11 June 2012; Accepted 12 June 2012

Academic Editor: Yi Su

Advanced MAC scheduling schemes provide efficient support of multimedia services in multiuser OFDM wireless networks. Designed in a cross layer approach, they opportunistically consider the channel state and are well adapted to the wireless multipath fading environment. These schedulers take advantage of time, frequency, and multiuser diversity. Thereby they maximize the global system throughput while ensuring the highest possible level of fairness. However their performances heavily depend on the bandwidth granularity (i.e., the number of elementary resource units) that is used in the resource allocation process. This paper presents and compares the main OFDM scheduling techniques. In particular it studies the influence of bandwidth granularity on the resource allocation strategies performances. The paper reveals that though bandwidth granularity has never been considered in former studies, it is of major importance for determining the application range of advanced OFDM scheduling techniques.

1. Introduction

Bandwidth allocation in next generation broadband wireless networks is a challenging issue. The schedulers shall provide mobile multimedia transmission services with an adequate Quality of Service (QoS). These new multimedia services with tight QoS constraints require increased system capacity together with fairness.

The past decades have witnessed intense research efforts on wireless communications. In contrast with wired communications, wireless transmissions are subject to many channel impairments such as path loss, shadowing, and multipath fading [1–4]. These phenomena severely affect the transmission capabilities and in turn the QoS experienced by applications, in terms of data integrity but also in terms of the supplementary delays or packet losses which appear when the effective bit rate at the physical layer is low. Among all candidate transmission techniques for broadband transmission, Orthogonal Frequency Division Multiplexing (OFDM) has emerged as the most promising physical layer technique for its capacity to efficiently reduce the harmful effects of multipath fading. This technique is already widely implemented in most recent wireless systems like 802.11a/g or 802.16. The basic principle of OFDM for fighting the effects of multipath propagation is to subdivide the available channel bandwidth in subfrequency bands of width inferior to the coherence bandwidth of the channel (inverse of the delay spread). The transmission of a high speed signal on a broadband frequency selective channel is then substituted with the transmission on multiple subcarriers of slow speed signals which are very resistant to intersymbol interference and subject to flat fading. This subdivision of the overall bandwidth in multiple channels provides frequency diversity. Added to time and multiuser diversity, this may result in a very spectrally efficient system subject to an adequate scheduling.

More recently intense research efforts have been given in order to propose efficient schedulers for OFDM-based wireless networks. Opportunistic schedulers, in particular, have received much attention [5–9]. These schemes preferably allocate the resources to the active mobile(s) with the most favourable channel conditions at a given time instant. This takes maximal profit of the multiuser diversity and frequency diversity in order to maximize the system throughput.

In fact, these schedulers highly rely on diversity for offering their good performances. Several studies analyze precisely the influence of the multiuser diversity on the performances of the reference schedulers [10–16]. However no work is focused on the role of frequency diversity which highly relies on bandwidth granularity. In this paper, we consider a fixed amount of bandwidth. Following the OFDM principle, this bandwidth is divided into subfrequency bands called subcarriers. In the allocation process, the different subcarriers are grouped for constituting the elementary resource units that are allocated to the mobiles. The higher the number of subcarriers in a group, the less the freedom and the (bandwidth) granularity in the allocation process and, consequently, the less the frequency diversity and the efficiency of the opportunistic resource allocation process. It is interesting to note that it is however not always interesting to maximize the bandwidth granularity. Indeed, the higher the bandwidth granularity is the more efficient are these schedulers, but it is at the expense of supplementary signaling overhead that can become prohibitive. A tradeoff has to be found.

In this paper, we study the influence of bandwidth granularity on the performances of the OFDM reference schedulers considering multiple metrics: system throughput, delay, jitter, fairness, and so forth. In contrast with multiuser diversity, the effect on the performances of bandwidth granularity is much more complex to analyze. Some papers compare the most acknowledged opportunistic schedulers but often show conflicting results. We show in this paper that this is mainly due to differences in bandwidth granularity hypotheses while each reference scheduler has its specificities and is built on specific assumptions. And, in particular, the performances of these schedulers strongly depend on the bandwidth granularity since, for example, some are primarily designed for single carrier communications, some for multicarrier systems. This paper clearly shows how a given scheduler can outperform another depending on the bandwidth granularity available in the system and why. We also provide in this paper the application range of each of the main OFDM reference schedulers with respect to bandwidth granularity.

The outline of the paper is as follows. Section 2 provides an overview of the OFDM system and a definition of bandwidth granularity. The main OFDM scheduling algorithms are described in the next section. Section 4 gives a detailed evaluation of the performance of these reference schedulers in a large range of bandwidth granularity values. Section 5 discusses the application range of each of the schedulers based on the results of Section 4. Section 6 concludes the paper.

2. System Description and Frequency Granularity

We consider Orthogonal Frequency Division Multiple Access (OFDMA). The physical layer is operated in TDD mode using the frame structure described in Figure 1 which ensures a good compatibility with existing systems like the OFDM-based transmission mode of the IEEE 802.16-2004 [17, 18]. The total available bandwidth is divided into subfrequency bands or subcarriers. The subcarrier spacing is constant and equal to a value inferior to the coherence bandwidth of the channel. Following [19], it is chosen in the order of 15 kHz. The total number of subcarriers is denoted n_{sub}. These subcarriers are grouped in n_g groups, with n_g a divider of n_{sub}. The scheduler may allocate any of these groups to any mobile. Consequently the higher the number n_g of groups, the higher the amount of *bandwidth granularity* and the higher the offered frequency diversity in the bandwidth allocation process.

The radio resource is further divided in the time domain in frames. Each frame is itself divided in time slots of constant duration. The time slot duration is an integer multiple of the OFDM symbol duration. Moreover, the frame duration is fixed to a value much smaller than the coherence time (inverse of the Doppler spread) of the channel. With these assumptions, the transmission on each subcarrier is subject to flat fading with a channel state that can be considered static during each frame. Transmissions performed on different subcarriers by different mobiles have independent channel state variations [20]. On each subcarrier, the modulation scheme is QAM with a modulation order adapted to the channel state between the access point and the mobile to which it is allocated. This provides the flexible resource allocation framework required for opportunistic scheduling. Note that the higher the bandwidth granularity n_g, the more flexible the allocation and the higher the diversity. The elementary resource unit (RU) is defined as any (group of subcarrier, time slot) pair. Each of these RUs may be allocated to any mobile.

3. Overview of the OFDM Reference Schedulers

We consider a centralized and synchronized approach [21]. The packets originating from the backhaul network are buffered in the access point which schedules the downlink transmissions (Figure 2). In the uplink, the mobiles signal their traffic backlog to the access point which builds the uplink resource mapping.

The MAC protocols currently used in wireless networks were originally and primarily designed for wired networks. These classical access methods like Round Robin (RR) and Random Access (RA) are not well adapted to the wireless environment and provide poor throughput. Consequently, much interest has recently been given to the design of scheduling schemes that maximize the performance of multiuser OFDM systems. In the following, we focus on the three major scheduling techniques that emerged: Maximum Signal-to-Noise Ratio (MaxSNR), Proportional Fair (PF), and Weighted Fair Opportunistic (WFO) scheduling.

3.1. Maximum Signal-to-Noise Ratio Scheduling. Many schemes are derived from the Maximum Signal-to-Noise Ratio (MaxSNR) technique (also known as Maximum Carrier-to-Interference ratio (Max C/I)) which allocates the resource at a given time to the active mobile with the greatest SNR [5, 6, 22]. Denoting $m_{k,n}$ the maximum number of bits that can be transmitted on a time slot of Resource Unit n if

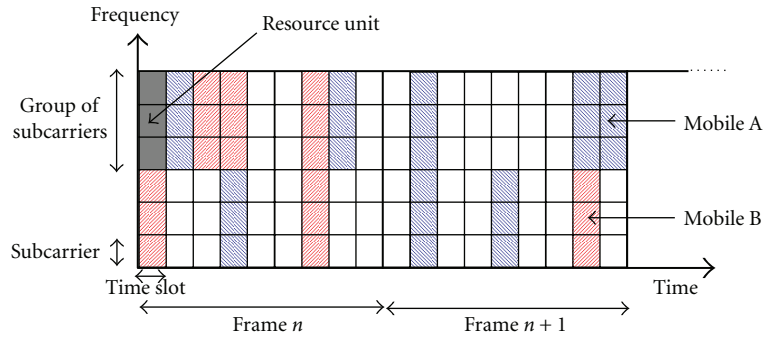


Figure 1: Frame structure in TDD mode ($n_{\text{sub}} = 6$ and $n_g = 2$).

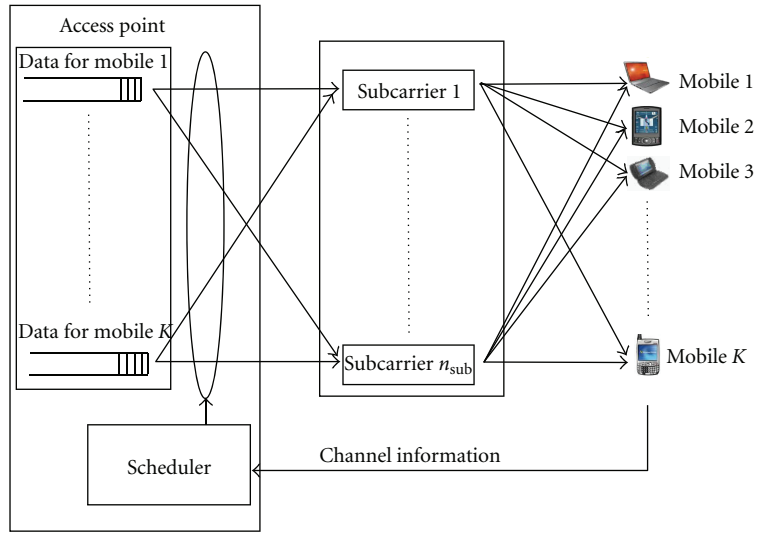


Figure 2: Opportunistic allocation of the radio resources among the set of mobiles in the downlink.

it is allocated to the mobile k, MaxSNR scheduling consists in allocating the RU n to the mobile j which has the greatest $m_{k,n}$ such as

$$j = \underset{k}{\operatorname{argmax}}(m_{k,n}), \quad k = 1, \ldots, K, \tag{1}$$

where K is the total number of mobiles.

Profiting of multiuser and bandwidth granularity, MaxSNR continuously allocates the radio resources to the mobile with the best spectral efficiency. Consequently, MaxSNR strongly increases the system throughput. Dynamically adapting the modulation and coding allows to always make the most efficient use of the radio resource and come closer to the Shannon limit. However MaxSNR assumes that the user with the most favourable transmission conditions has information to transmit at the considered time instant. It does not take into account the variability of the traffic and the queuing aspects. Additionally, a negative side effect of this strategy is that the closest mobiles to the access point have disproportionate priorities over mobiles more distant since their path loss attenuation is much smaller. This results in a severe lack of fairness.

3.2. Proportional Fair Scheduling. Proportional Fair (PF) algorithms have recently been proposed to incorporate a certain level of fairness while keeping the benefits of multiuser diversity [7, 8, 23–26]. The basic principle is to allocate resources to a mobile j when its channel conditions are the most favourable with respect to its time average such as

$$j = \underset{k}{\operatorname{argmax}}\left(\frac{m_{k,n}}{M_{k,n}}\right), \quad k = 1, \ldots, K, \tag{2}$$

where $M_{k,n}$ is the time average of the $m_{k,n}$ values.

At a short time scale, path loss variations are negligible and channel state variations are mainly due to multipath fading, statistically similar for all mobiles. Thus, PF provides an equal sharing of the total available bandwidth among the mobiles as RR. Applying an opportunistic scheduling approach, the system throughput increase is comparable to the MaxSNR gain [9]. Actually, PF combines the advantages of the classical schemes and the opportunistic schemes. It currently appears as the best bandwidth management scheme.

In PF-based schemes, fairness consists in guaranteeing an equal share of the total available bandwidth to each mobile, whatever its position or channel conditions. However, since the farther mobiles have a lower spectral efficiency than the closer ones due to pathloss, all mobiles do not all benefit of an equal average throughput despite they all obtain an equal share of bandwidth. This induces heterogeneous delays and unequal QoS. [12, 23, 27] demonstrate that fairness issues persist in PF-based protocols when mobiles have unequal spatial positioning, different traffic types, or different QoS targets. In particular PF scheduling does not take into account the delay constraints and is not well adapted to multimedia services which introduce heterogeneous users, new traffic patterns with highly variable bit rates and stringent QoS requirements in terms of delay and packet loss.

3.3. Weighted Fair Opportunistic Scheduling. More recently a new MAC scheduler, called "Weighted Fair Opportunistic (WFO)," has been proposed for efficient support of multimedia services in multiuser OFDM wireless networks [9, 28]. Built in a higher layers/MAC/PHY cross-layer approach, this scheme is designed for best profiting of the multiuser diversity and taking advantage of the dynamics of the multiplexed traffics. It takes into account both the transmission conditions and the higher layer constraints (traffic patterns, QoS constraints). In order to provide an efficient support of multimedia transmission services, WFO dynamically favors the mobiles that go through a critical period with respect to their QoS requirements using dynamic priorities.

Evaluating if a mobile goes through a critical period should not only focus on the classical mean delay and jitter analysis. Indeed, a meaningful constraint regarding delay is the limitation of the occurrences of large values. Accordingly, [9] defines the concept of *delay outage* by analogy with the concept of outage used in system coverage planning. A mobile k is in delay outage (in critical period) when its packets experience a delay greater than a given threshold T_k defined by the mobile application requirements. The delay experienced by each mobile is tracked all along the lifetime of its connection. At each transmission of a packet of mobile k, the ratio of the total number of packets whose delay exceeded the threshold divided by the total number of packets transmitted since the beginning of the connection is computed. The result is called Packet Delay Outage Ratio (PDOR) of mobile k and is denoted PDOR_k. This measure is representative of the emergency for the mobile k to be served. Figure 3 illustrates an example cumulative distribution of the packet delay of a mobile at a given time instant. A mobile can be considered as satisfied when, at the end of its connection, its delay constraint is met, that is, its experienced PDOR is less than a PDOR target specific to the mobile application.

The WFO scheduling principle is to allocate a Resource Unit n to the mobile j which has the greatest WFO parameter value $\mathrm{WFO}_{k,n}$ with

$$j = \underset{k}{\operatorname{argmax}}(\mathrm{WFO}_{k,n}), \quad k = 1, \ldots, K, \tag{3}$$

where $\mathrm{WFO}_{k,n}$ is equal to

$$\mathrm{WFO}_{k,n} = m_{k,n} \times f(\mathrm{PDOR}_k), \tag{4}$$

with f a strictly increasing polynomial function defined in [9].

With this scheduling, physical layer information (represented through the factor $m_{k,n}$) are used in order to take advantage of the time, frequency and multiuser diversity and maximize the system capacity. Higher layer information (represented through the factor $f(\mathrm{PDOR}_k)$) are exploited in order to introduce dynamic priorities between the flows for ensuring the same QoS level to all mobiles. With this original weighted system that introduces dynamic priorities between the flows, WFO keeps a maximum number of flows active across time but with relatively low traffic backlogs. This results in a well-balanced resource allocation. Preserving the multiuser diversity allows to continuously take a maximal benefit of opportunistic scheduling and thus maximize the bandwidth usage efficiency. When the bandwidth granularity is sufficient, WFO better conceals the system capacity maximization, QoS support and fairness objectives than PF and MaxSNR as will be observed in the next section.

4. Performance Evaluation

In this section, we study the allocation of radio resources among the set of mobiles situated in the coverage zone of an access point using the classical Round Robin (RR) and the most efficient opportunistic schedulers presented in Section 3: MaxSNR, PF, and WFO. Performance evaluation results are obtained using OPNET discrete event simulations. We focus on five essential performance criteria:

(i) offered system capacity;

(ii) mean buffer occupancy;

(iii) delay and jitter;

(iv) perceived QoS satisfaction level.

4.1. Simulation Setup. In the simulations, n_{ts} is defined as the number of time slots available in a frame. We assume a total number n_{sub} of 128 subcarriers and a total number n_{ts} of 5 time slots in a frame. In addition, the frame duration is fixed to 2 ms so that the channel state can be considered static during each frame. The results are analyzed with respect to different bandwidth granularities: n_g ranges between 1 and 128 which, respectively, corresponds to groups of 128 subcarriers to 1 subcarrier.

A detailed description of the channel model is given in [28]. The channel gain model on each subcarrier assumes free space path loss and multipath Rayleigh fading [4]. We consider a reference distance d_{ref} for which the free space attenuation equals a_{ref}. As a result the channel gain is given by

$$a_{k,n} = a_{\mathrm{ref}} \times \left(\frac{d_{\mathrm{ref}}}{d_k}\right)^{3.5} \times \alpha_{k,n}^2, \tag{5}$$

where d_k is the distance to the access point of the mobile owning the service flow k and $\alpha^2_{k,n}$ represents the flat fading experienced by this service flow k if transmitted on the RU n. In the following, $\alpha_{k,n}$ is Rayleigh distributed with an expectancy equal to unity. The maximum transmit power satisfies

$$10\log_{10}\left(\frac{P_{\max}T_s}{N_0}\times a_{\text{ref}}\right) = 31\,\text{dB}. \quad (6)$$

Additionally, the BER target is taken equal to 10^{-3}.

We consider 8 mobiles which run a videoconference application with successive connections of five minutes duration. The traffic is composed of an MPEG-4 video stream [29] multiplexed with an AMR voice stream [30]. This demanding type of application, typical of multimedia, generates a high volume of data with high sporadicity and requires tight delay constraints which substantially complicates the task of the scheduler. The average bit rate of each source is 80 Kbps. The PDOR target is set to 5% and the delay threshold T_k is fixed to the value of 80 ms considering real time constraints.

As described in Section 3, the main performance differences between MaxSNR, PF, and WFO regards fairness with respect to mobiles located at unequal positions. This paper proposes to study the influence of the frequency granularity on the performances of the schedulers. Though never studied before, this factor has a very strong impact on the performances of the schedulers. In order to facilitate the analysis, we have chosen to represent only the results of the simulations that involve static users. Simulations with mobile users have also been carried out but when mobility is considered, the results are much more complex to analyze with respect to bandwidth granularity. The mobility model actually interferes with the performances of the schedulers and the role of the bandwidth granularity then could not clearly be isolated. Consequently, in the presented simulation results and in order to observe how fairness is affected by the amount of bandwidth granularity available in the system, a first half of mobiles are situated close to the access point (2.5 km) so that their mean $m_{k,n}$ value equals 3 bits per subcarrier and per time slot. The second half of the mobiles are situated twice over farther (5 km).

4.2. Qualitative Impact of Frequency Granularity on the Scheduling Performances. The performances of the OFDM reference schedulers described in Section 3 depend on the amount of bandwidth granularity available in the system. Each scheduler is more or less sensible to an increase of the bandwidth granularity.

Figure 4 shows the distribution of the number of bits that may be transmitted per subcarrier and per time slot by far and close mobiles, considering the channel model described in [28]. In Figure 4(a), the bandwidth granularity is low: n_g equals 1, that is, the system accommodates only 1 group of 128 subcarriers. Figure 4(b) corresponds to the highest bandwidth granularity: n_g equals 128, that is, there are 128 groups of 1 subcarrier.

Note that the number of bits that may be transmitted on a subcarrier can only be a discrete number. Indeed, in the system under study, the elementary data unit is a bit. Hence, during a time slot, only an integer number of bits can be transmitted on a subcarrier. This explains why when n_g equals 128 (128 groups of 1 subcarrier), the distribution of the number of bits that may be transmitted on a subcarrier is constituted only with discrete and integer values (as shown in Figure 4(b)). Similarly, when n_g equals 1, the scheduler has only one capacious Resource Unit (RU) to allocate. This RU is constituted by one group of 128 subcarriers that may be allocated to a single mobile only. As in Figure 4(b), the total number of bits that is then transmitted on the 128 subcarriers during a timeslot remain integer and discrete but are much higher since they relate globally to the 128 subcarriers. In order to compare the results on a same basis, we computed in Figure 4(a) the number of bits that may be transmitted on average per subcarrier when n_g equals 1 (and not on a group of subcarriers).

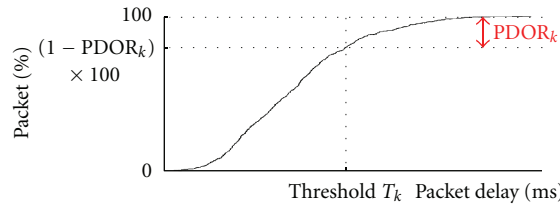
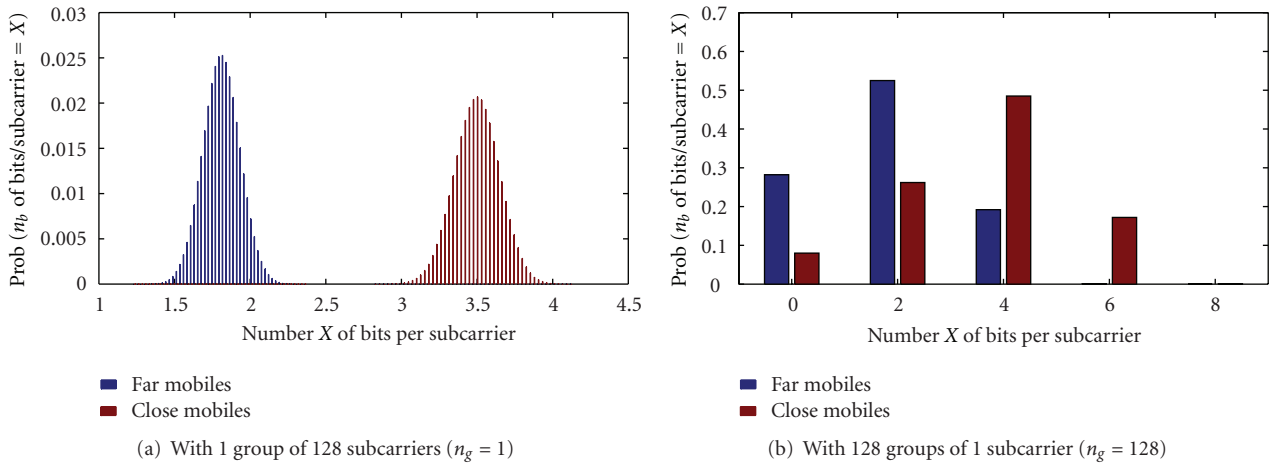


Figure 3: Example packet delay CDF and experienced PDOR.

Figure 4(a) shows that the distributions of the number of bits that may be transmitted on a subcarrier for far and for close mobiles when n_g equals 1 are very narrow compared to the case where n_g equals 128 (Figure 4(b)). Indeed, when the subcarriers are grouped, the quality fluctuations of the subcarriers compensate each other inside the group. As a consequence of this averaging effect, the capacity of a group of subcarriers is pretty much constant which in turn offers less degrees of freedom in the resource allocation at system level. This is why the performances worsen in terms of capacity.

Additionally, we observe in Figure 4(a) that, in contrast with Figure 4(b), the distributions obtained for the two groups of mobiles have no intersection. When all subcarriers are grouped, the mean number of bits that may be transmitted by close mobiles over a group of subcarriers is always larger than the transmit possibility of far mobiles due to the averaging effect. This results in a really unfair allocation for MaxSNR-based schedulers when all subcarriers are grouped since close mobiles then have absolute priority over far mobiles. In contrast, when subcarriers may be allocated individually, close mobiles may get a chance at some time instant to gain access to a subcarrier when its fading conditions are favorable on this subcarrier. This is why the higher the value of n_g, the lower the unfairness of the MaxSNR-based scheduler.

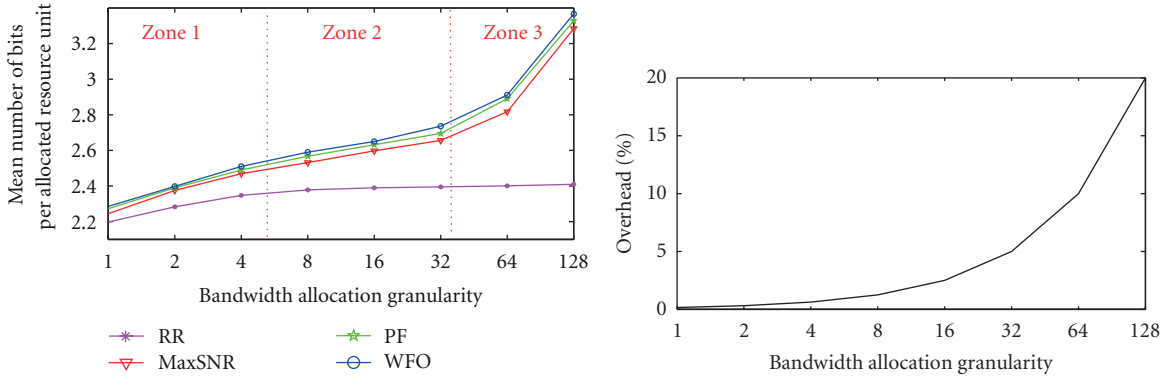
4.3. System Capacity. We first studied the system capacity offered by the four scheduling algorithms as a function of the bandwidth granularity. Figure 5 analyses the impact of



(a) With 1 group of 128 subcarriers ($n_g = 1$)

(b) With 128 groups of 1 subcarrier ($n_g = 128$)

FIGURE 4: Number of bits per subcarrier considering two distinct bandwidth granularities.



(a) Spectral efficiency per RU

(b) Overhead

FIGURE 5: Impact of bandwidth granularity on spectral efficiency and overhead.

a bandwidth granularity increase on spectral efficiency and overhead.

The first effect is a spectral efficiency increase as shown in Figure 5(a) which gives the average number of bits carried on a time slot of a used subcarrier by each tested scheduler under diverse bandwidth granularities. As expected, RR does not take benefit of multiuser diversity which results in a bad utilization of the bandwidth and, in turn, poor system throughput. Based on opportunistic scheduling, the three other schemes globally show better performances. Improving the spectral efficiency of the system, less RUs are needed than with RR for managing the same amount of traffic, and the system capacity increases.

The system capacity of a wireless network actually highly relies on the level of spectral efficiency that is provided by the scheduler. The purpose of the opportunistic schedulers is to allocate the radio resources at a given instant to the mobiles that have the best radio conditions and have data to transmit at that time instant. If the considered mobile does not have data to transmit, the resource unit is allocated to another mobile whose radio conditions are worse. Consequently the spectral efficiency decreases. Thus, multiuser diversity plays a central role for the maximization of the system capacity. The specificity of the opportunistic schedulers is that they take benefit of the multi-user, time and frequency diversity in order to increase this spectral efficiency and consequently the system capacity.

For each scheduler, we observe an interesting inflection of the curve when the bandwidth granularity increases resulting from the superimposition of the following three phenomena.

(i) The first is predominant mainly in Zone 1 and affects all the schedulers. We observe a spectral efficiency increase which is due to the sparing of many subcarriers used for the transmission of only few bits. To be clearer, let us consider that a given mobile has only a few bits to transmit. If n_g equals 1, the scheduler has only capacious Resource Units to allocate. As a consequence, many subcarriers are allocated (128 subcarriers if n_g equals 1) while maybe

one or two could be sufficient. Consequently, the excess of allocated subcarriers in the group is wasted while with more frequency granularity they could be allocated to another mobile. This yields a huge waste of resource. The elementary shares of resource are too large. With a value of n_g of 2, the scheduler allocates 2 groups of 64 subcarriers and this waste is reduced by approximately 50%. With a value of n_g of 4, it is reduced by about 75%, and so forth. Thus, an interesting capacity gain may be obtained with only a little more bandwidth granularity. Sufficient granularity is then provided in the scheduling for allocating an amount of resource that quite closely matches the need of the mobile.

(ii) In conjunction to this, a second phenomenon affects the curves of the opportunistic schedulers only. This phenomenon is clearly visible in Zone 2 and 3 where the curve of the RR is nearly flat as opposed to the curves of the opportunistic schedulers which exhibit a sensible inflection. This shows that they take advantage of the supplementary bandwidth granularity when n_g increases. As explained in Section 4.2, when the bandwidth granularity increases, the distribution values of the number of bits that may be transmitted on an RU spreads out. With more available combination choices in the bandwidth allocation, MaxSNR, PF, and WFO increase their system capacity gain.

(iii) The third phenomenon dominates in Zone 3 mainly, accentuating the inflection of the curve of the opportunistic schedulers. When a group of subcarriers is allocated, some subcarriers of this group may go unused because their quality is very poor. They are nevertheless wasted because they belong to the allocated group. This waste totally disappears when the bandwidth granularity is maximal. Indeed, when n_g equals 128, if a subcarrier is not usable by a given mobile, it is spared for another mobile. As a general rule, a higher bandwidth granularity helps to avoid transmitting no bit at all on some subcarriers of a group. This results in a higher spectral efficiency.

The second effect of a bandwidth granularity increase is a greater signaling overhead. Figure 5(b) represents the price to pay in terms of signaling as a function of the bandwidth granularity. Indeed, for each allocated group of subcarriers, the scheduler must signal which mobile is chosen. Consequently, when n_g increases, a greater part of the bandwidth is wasted. Indeed, denoting n_s the number of (subcarrier, time slot) pairs needed to transmit this information, the overhead ratio can be defined as

$$\rho = \frac{n_g \times n_s}{n_f \times n_{\text{sub}} \times n_{ts}}, \tag{7}$$

where n_f is the number of frames allocated during one scheduling step. We considered that the scheduling process allocates the RUs frame by frame. Consequently n_f is set to 1. Additionally, we assume that n_s can be set to 1 since the information to signal can be represented by a small integer (since n_{sub} is set to 128 and n_{ts} is set to 5).

Figure 6 combines the results of Figures 5(a) and 5(b). It presents the bandwidth usage ratio that is defined as the mean number of allocated RUs divided by the total number of RUs in the frame, including both the useful data and the signaling overhead. Considering only the results of Figure 5(a), we could expect that the higher the bandwidth granularity, the greater the spectral efficiency, the lower the bandwidth usage ratio, and the greater the remaining capacity for accommodating other potential users. However, the supplementary signaling overhead cannot be ignored and Figure 6 shows that, for each scheduler, a high bandwidth granularity n_g induces too much overhead which can not be compensated by a better frequency diversity use. These results yield the application range of each scheduler. RR's best performances are obtained when n_g equals 8, MaxSNR and PF provide their best performances when n_g equals 16 and WFO performances which strongly rely on diversity reach their optimum when n_g equals 32.

The bandwidth usage ratio results have been collected through simulation. However, it is possible to relate the spectral efficiency and overhead ratio to the bandwidth usage ratio as follows. The bandwidth usage ratio is equal to the total average bit rate of data to transmit divided by the system capacity. In our simulation, we use 8 sources with an average bit rate of 80 Kbps. This corresponds to a total average bit rate of 640 Kbps. Let us denote n_{bRU} as the mean number of bits per allocated RU (Figure 5(a)), ρ the overhead ratio (Figure 5(b)), and n_{fps} the number of frames to allocate per second. The system capacity is

$$C = \left(n_{\text{sub}} \times n_{ts} \times n_{fps}\right) \times n_{\text{bRU}} \times (1 - \rho). \tag{8}$$

For example, with RR, when n_g equals 1, the bandwidth usage ratio is

$$\frac{640}{(128 \times 5 \times 500) \times 2.2 \times (1 - 0.0015625)} = 91.1\%. \tag{9}$$

As another example, with MaxSNR, when n_g equals 32, the bandwidth usage ratio is

$$\frac{640}{(128 \times 5 \times 500) \times 2.65 \times (1 - 0.05)} = 79.4\%. \tag{10}$$

Note that these theoretical values are very close to the simulation results values presented in Figure 6. This confirms the correctness of the simulation results.

4.4. Buffer Occupancy, Packet Delay, and Jitter. Figures 7(a), 7(b), and 7(c), respectively, show the mean buffer occupancy, the mean packet delay, and the mean packet jitter (RR performances are not presented in Figures 7(a) and 7(c) since its performances are not in the same order of magnitude. RR is not competitive here with the opportunistic schedulers). Section 4.3 points out the three phenomena that yield a higher system capacity when the bandwidth granularity increases. This gain in system capacity has a direct impact on the system performances in terms of mean buffer occupancy, delay, and jitter as long as the overhead impact is not too important. As expected, when n_g increases until the

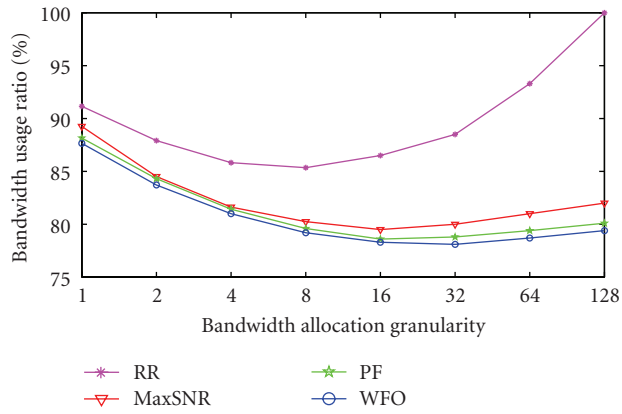


Figure 6: System capacity analysis.

optimum value cited in previous subsection, these three performance parameters (buffer occupancy, delay, jitter) improve for each scheduler thanks to a better spectral management since the gain due to supplementary frequency diversity is greater than the generated overhead. Additionally, it is important to note that the QoS improvement is very quick when the bandwidth granularity is incremented from 1 to 2. Afterwards, the gain is limited for all opportunistic schedulers. This shows that it is very advantageous to offer at least a few bandwidth granularity for improving these performance metrics.

4.5. Fairness and Mobile Satisfaction. Fairness is the most difficult objective to reach. It consists in ensuring the same QoS to all mobiles, whatever their position. In Figure 8, we examine the mobile dissatisfaction in terms of delay, discriminating with respect to the mobile position. We consider a mobile is dissatisfied when, at the end of its connection, its delay constraint is not met, that is, its experienced PDOR is greater than its PDOR target. So, the mobile dissatisfaction ratio is the percentage of mobile connections that do not satisfy the PDOR target. It is interesting to note that the four tested schedulers have not the least the same behavior when the amount of bandwidth granularity available in the system increases.

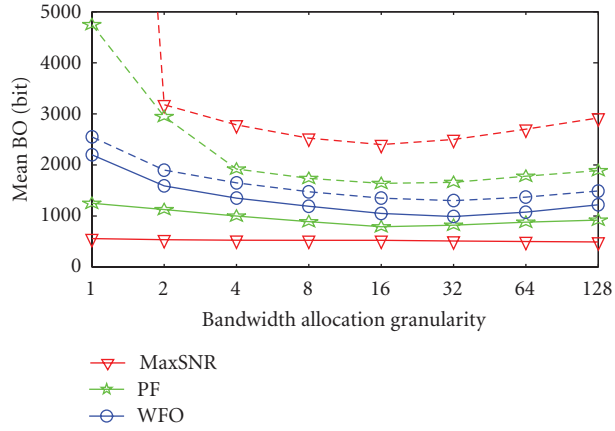
Classical RR yields bad results. Indeed, since multiuser diversity is not exploited, the overall spectral efficiency is small and system throughput is low. Consequently, the delay targets are widely exceeded and mobiles are dissatisfied. Naturally the spectral efficiency improves as the bandwidth granularity increases (Figure 5(a)). When the value of n_g is maximum, all schedulers succeed to transmit more bits on a subcarrier than when n_g is lower. However, the price to pay for a bandwidth granularity increase is a high overhead ratio growth. A bandwidth granularity increase is not beneficial to all the schedulers. For example, when n_g equals 128, only 80% of the capacity stays available for managing the useful data traffic (Figure 5(b)). In the case of RR, the gain provided by increasing n_g from 64 to 128 is negligible (Figure 5(a)) compared to the high overhead ratio increase (Figure 5(b)). This induces a system overload. As shown in Figure 6, 100% of the system RUs are used but much more are required. Indeed, the number of packets waiting for transmission builds up in the system, the packet delay values explode (Figure 7(b)), and, consequently, the users cannot be satisfied since their QoS requirements are not fulfilled (Figure 8). In conclusion, the results regarding mobile satisfaction (Figure 8) are highly related to the bandwidth usage ratio results (Figure 6), themselves depending on the joint results of spectral efficiency and overhead (Figures 5(a) and 5(b)).

Highly unfair, MaxSNR fully satisfies the required QoS of close mobiles at the expense of the satisfaction of far mobiles. Unnecessary priorities are given to close mobiles who easily respect their QoS constraints while more attention should be given to the farther. This inadequate priority management dramatically increases the global mobile dissatisfaction. Focusing on results of Figure 8(c), we observe that the mobile satisfaction slowly increases at first then decreases with the addition of bandwidth granularity. Indeed, opportunistic schedulers intrinsically need diversity. The higher the bandwidth granularity in the system is the more the MaxSNR scheduler takes advantage of it. Improving the spectral efficiency, the mobile satisfaction is directly impacted as far as the cost in terms of overhead is not too large. PF is subject to the same tendency and is even more reactive.

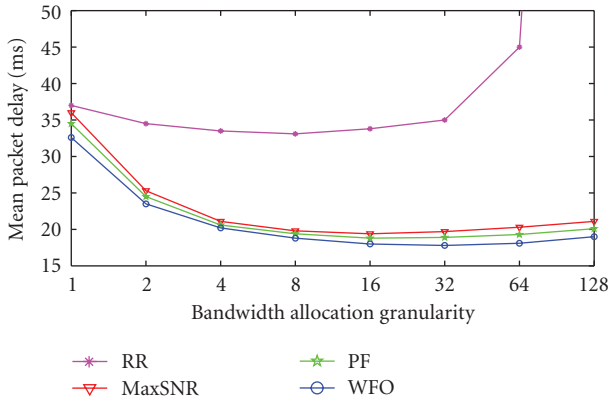
In contrast, with WFO, the easy satisfaction of close mobiles (with better spectral efficiency) offers a degree of freedom which is exploited in order to help the farther ones. WFO dynamically adapts the priorities as a function of the mobile position. This results in a fair allocation of the RUs providing the same level of QoS for each group of mobiles whatever their respective position. However, WFO needs more bandwidth granularity than the other schedulers for well developing its abilities. If we focus on the global mobile dissatisfaction (Figure 8(c)), with low bandwidth granularity (n_g equals 1), all the other tested schedulers offer a greater number of satisfied mobiles. But, this supposes the sacrifice of the far mobiles in order to guarantee the QoS constraints of close mobiles. When n_g equals 1, WFO has not the means to an end. However, in contrast with the other schemes, as soon as n_g increases, WFO takes benefit of the supplementary diversity and very quickly improves the QoS provided to all mobiles, widely outperforming the other schedulers.

5. Discussion

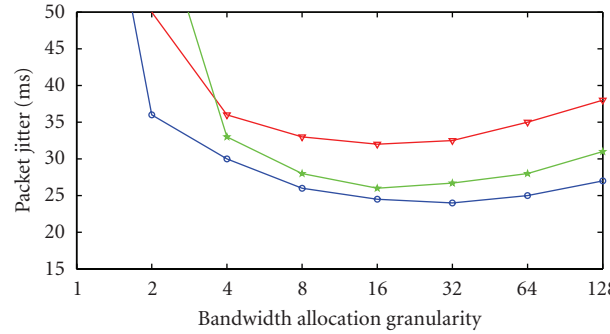
It appears that the performance of the considered schedulers in terms of system capacity, delay, jitter, fairness, QoS support, and so forth is widely different. As underlined in Figures 4, 6, 7, and 8 each scheduler is *bandwidth granularity dependent* and does not have the same behavior as a function of the bandwidth granularity variation. The respective performance of the studied schedulers also changes with the bandwidth granularity context. Additionally, Section 4 showed that it differs depending on the studied metrics too. For instance, PF outperforms WFO in terms of mobile satisfaction when n_g equals 1 but the situation is different if we focus on the mean delay, jitter, or spectral efficiency. Moreover, regarding mobile satisfaction, when n_g increases



(a) Mean buffer occupancy for close mobiles (solid lines) and far mobiles (dashed lines)



(b) Mean packet delay



(c) Mean packet jitter

Figure 7: Buffer occupancy, delay, and jitter.

and reaches 4, WFO outperforms PF. According to the results presented in this paper, it seems really important to have a precise knowledge of the available bandwidth granularity in the system for choosing the scheduler. With a low bandwidth granularity, the opportunistic schedulers do not provide a large gain compared to the classical Round Robin scheduling which is much less complex. On the contrary, the higher the bandwidth granularity, the more interesting is the use of evolved schedulers. Indeed, a low bandwidth granularity highly degrades the advantages provided by each opportunistic scheduler. If we focus on the spectral efficiency, Figure 5(a) shows that for a value of n_g running from 128 to 1, the mean number of bits transmitted per allocated RU, respectively, decreases from 3.37 to 2.2 which corresponds to a fall by one-third.

In addition, this study demonstrates that, whatever the considered scheduling, full sharing of the bandwidth in many groups of subcarriers is not profitable for the system. Indeed, for each scheduler a tradeoff exists for a bandwidth granularity n_g ranging between 8 and 32. Higher n_g values provide a too large overhead.

6. Conclusion

In this paper, we have compared the major resource allocation strategies for OFDM wireless networks. In addition, we have isolated the bandwidth granularity factor in order to study its influence on the inherent multiplexing gain offered by the OFDM schedulers. Comparing them for different bandwidth granularity values, this study has underlined that each scheduler is highly bandwidth granularity dependent and that performances are strongly related to this factor. When the bandwidth granularity increases, the performance variations depend on the studied performance metrics and the type of scheduler. Each scheduler have a limited application range with respect to the focused metrics. For instance, regarding delay and jitter, WFO seems to offer the best solution whatever the bandwidth granularity available while in terms of mobile satisfaction it outperforms PF and MaxSNR only in a context where bandwidth granularity is reasonable. Moreover jointly focusing on all performance metrics, the RR scheduler with low complexity is not too bad in a low bandwidth granularity context while MaxSNR,

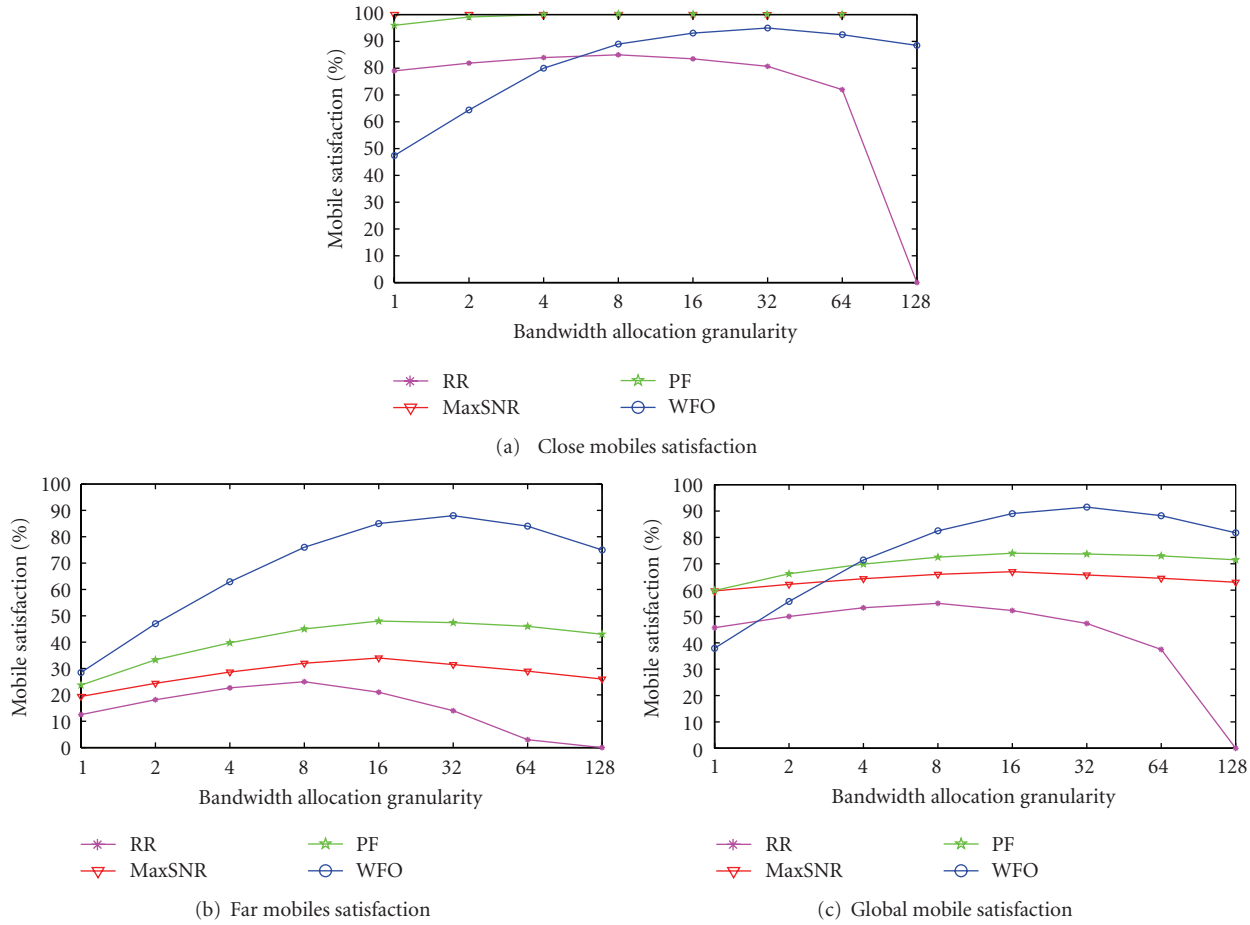


Figure 8: Analysis of the mobile satisfaction.

PF, and WFO are mainly designed for operating in highly frequency diversified systems.

References

[1] J. G. Proakis, *Digital Communications*, McGraw-Hill, New York, NY, USA, 3rd edition.

[2] R. Steele and L. Hanzo, *Mobile Communications*, IEEE Computer Society Press, 2000.

[3] A. Goldsmith, *Wireless Communications*, Cambridge University Press, 2005.

[4] J. D. Parsons, *The Mobile Radio Propagation Channel*, Wiley, 1992.

[5] R. Knopp and P. A. Humblet, "Information capacity and power control in single-cell multiuser communications," in *Proceedings of the IEEE International Conference on Communications*, vol. 1, pp. 331–335, June 1995.

[6] C. Y. Wong, R. S. Cheng, K. B. Letaief, and R. D. Murch, "Multiuser OFDM with adaptive subcarrier, bit, and power allocation," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 10, pp. 1747–1758, 1999.

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

[8] H. Kim, K. Kim, Y. Han, and S. Yun, "A proportional fair scheduling for multicarrier transmission systems," in *Proceedings of the IEEE 60th Vehicular Technology Conference (VTC '04)*, vol. 2, pp. 409–413, September 2004.

[9] C. Gueguen and S. Baey, "Scheduling in OFDM wireless networks without tradeoff between fairness and throughput," in *Proceedings of the 68th Semi-Annual IEEE Vehicular Technology (VTC '08)*, September 2008.

[10] H. Zhou, D. Yang, W. Qi, and M. Ma, "On performance of mutiuser diversity in SISO and MIMO wireless communication," in *Proceedings of the 14th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '03)*, vol. 3, pp. 2872–2876, September 2003.

[11] M. Sánchez-Fernández, M. Luz Pablo-González, and A. Lozano, "Exploiting multiuser diversity through uplink scheduling," in *Proceedings of the IEEE 61st Vehicular Technology Conference (VTC '05)*, vol. 3, pp. 1918–1920, June 2005.

[12] J. G. Choi and S. Bahk, "Cell-throughput analysis of the proportional fair scheduler in the single-cell environment," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 766–7778, 2007.

[13] K. Kansanen and R. R. Müller, "Multiuser diversity in channels with limited scatterers," in *Proceedings of the IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '08)*, pp. 1–5, September 2008.

[14] T. W. Ban, W. Choi, B. C. Jung, and D. K. Sung, "Multi-user diversity in a spectrum sharing system," *IEEE Transactions on Wireless Communications*, vol. 8, no. 1, pp. 102–106, 2009.

[15] X. Zhang, W. Wang, and X. Ji, "Multiuser diversity in multi-user two-hop cooperative relay wireless networks: system model and performance analysis," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 1031–1036, 2009.

[16] A. Tajer and X. Wang, "Multiuser diversity gain in cognitive networks with distributed spectrum access," in *Proceedings of the 43rd Annual Conference on Information Sciences and Systems (CISS '09)*, pp. 135–140, March 2009.

[17] IEEEStd802.16-2004, "IEEE standard for local and metropolitan area networks, part 16: air interface for fixed broadband wireless access systems," IEEE Std 802. 16-2004, 2004.

[18] C. Hoymann, "Analysis and performance evaluation of the OFDM-based metropolitan area network IEEE 802.16," *Computer Networks*, vol. 49, no. 3, pp. 341–363, 2005.

[19] C. B. Ribeiro, K. Hugl, M. Lampinen, and M. Kuusela, "Performance of linear multi-user MIMO precoding in LTE system," in *Proceedings of the 3rd International Symposium on Wireless Pervasive Computing (ISWPC '08)*, pp. 410–414, May 2008.

[20] 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. 150–153, 2001.

[21] J.-J. Van De Beek, P. O. Börjesson, M.-L. Boucheret et al., "Time and frequency synchronization scheme for multiuser OFDM," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 11, pp. 1900–1914, 1999.

[22] X. Wang and W. Xiang, "An OFDM-TDMA/SA MAC protocol with QoS constraints for broadband wireless LANs," *Wireless Networks*, vol. 12, no. 2, pp. 159–170, 2006.

[23] C. Gueguen and S. Baey, "Compensated proportional fair scheduling in multiuser OFDM wireless networks," in *Proceedings of the 4th IEEE International Conference on Wireless and Mobile Computing, Networking and Communication (WiMob '08)*, pp. 119–125, October 2008.

[24] H. Kim, K. Kim, Y. Han, and J. Lee, "An efficient scheduling algorithm for QOS in wireless packet data transmission," in *Proceedings of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC' 02)*, vol. 5, pp. 2244–2248, September 2002.

[25] A. Wang, L. Xiao, S. Zhou, X. Xu, and Y. Yao, "Dynamic resource management in the fourth generation wireless systems," in *Proceedings of the International Conference on Communication Technology (ICCT '03)*, vol. 2, pp. 1095–1098, April 2003.

[26] P. Svedman, S. K. Wilson, and B. Ottersten, "A QoS-aware proportional fair scheduler for opportunistic OFDM," in *Proceedings of the IEEE 60th Vehicular Technology Conference (VTC '04)*, vol. 1, pp. 558–562, September 2004.

[27] J. M. Holtzman, "Asymptotic analysis of proportional fair algorithm," in *Proceedings of the 12th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '01)*, vol. 2, pp. F33–F37, October 2001.

[28] C. Gueguen and S. Baey, "An efficient and fair scheduling scheme for multiuser OFDM wireless networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 1610–1615, April 2008.

[29] S. Baey, "Modeling MPEG4 video traffic based on a customization of the DBMAP," in *Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS '04)*, July 2004.

[30] P. Brady, "A model for generating on-off speech patterns in two-way conversation," *Bell System Technical Journal*, vol. 48, 1969.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 169026, 10 pages
doi:10.1155/2012/169026

Research Article

Resource Management in Satellite Communication Systems: Heuristic Schemes and Algorithms

Shahaf I. Wayer and Arie Reichman

School of Engineering, Ruppin Academic Center, Emek Hefer, Israel

Correspondence should be addressed to Arie Reichman, arier@ruppin.ac.il

Received 26 February 2012; Revised 30 April 2012; Accepted 1 May 2012

Academic Editor: Yi Su

The high cost of frequency bandwidth in satellite communication emphasizes the need for good algorithms to cope with the resource allocation problem. In systems using DVB-S2 links, the optimization of resource allocation may be related to the classical multi-knapsack problem. Resource management should be carried out according to the requests of subscribers, their priority levels, and assured bandwidths. A satisfaction measure is defined to estimate the allocation processes. Heuristic algorithms together with some innovative scaling schemes are presented and compared using Monte Carlo simulation based on a traffic model introduced here.

1. Introduction

1.1. DVB-S2 in Satellite Communication. The most popular standard today in satellite communication is DVB-S2 [1], designed for transmission of digital television broadcasting. This relatively new standard is gradually replacing the classic DVB-S [2] in new systems as well as in existing systems. It is also used in new VSAT systems in forward link as well as in reverse link, in cases when the required data rates are higher than those supplied by DVB-RCS [3]. DVB-S2 is also used in satellite modems for point-to-point applications. In this paper we focus on a system that consists of DVB-S2 links with dynamic changes in the required parameters of the links due to changes in the traffic demands.

1.2. Resource Management. In order to save the operational cost of the satellite communication system, transmissions that share the total available frequency band have to be fairly divided according to the requirements of the subscribers [4]. For a star-type system with a central HUB that communicates with subscriber terminals, the control is conducted by the control center in the HUB according to subscriber needs. For other type of systems, like point-to-point, the control is conducted by a dedicated communication and control system that operates in addition to point-to-point links.

Regardless of the way the control is performed, a resource allocation algorithm has to be designed for optimal performances and best satisfaction.

In commercial systems run by a service provider, the Service Level Agreement (SLA), which is a set of parameters that defines the capabilities and the paid rights of every terminal in the system, connects the service provider and the subscribers. The Committed Information Rate (CIR) is the rate that is guaranteed by the system. Typically it is low enough so that the system can guarantee the user of this specific rate. The Peak Information Rate (PIR), on the other hand, is the maximum rate that may be allocated. When the user asks for a rate higher than CIR and lower or equal to PIR, the system will allocate this rate if available. When a user asks for a rate higher than PIR, the system will consider it as if the request was equal to PIR. The service provider has difficulty in providing the Committed Rate with certainty. In this research, we assumed that the bandwidth of a terminal is proportional to its rate. We use the term *assured bandwidth* as a quantity that the system can provide *most of the time* but not always.

The terminals have priority levels that determine their significance according to their payment to the service provider. In a situation of low resources, requests from high priority terminals may cause the system to decrease the rate of low priority terminals to their minimum.

In the course of the resource allocation cycle, subscribers issue requests for bandwidth. The bandwidths they own are removed from the spectrum list. The resource allocation process follows, and the bandwidths are reconfigured over the spectrum. This whole process of removing spectrum bands and reconfiguring the spectrum with the newly allocated bands creates gaps (*holes*) in the spectrum, resulting in a fragmented spectrum. The task of the resource allocation algorithms is to fit the requests into those *holes* in the spectrum.

In [5] it was shown that this task is equivalent to the multi-knapsack problem (MKP). Two heuristic algorithms were presented together with a single scaling scheme for the bandwidth requests. Scaling down the requests is necessary when their sum exceeds the available bandwidth budget, and utilizing the scaling scheme improves the algorithm results.

In this paper, we review the MKP problem and suggest an extension to it. We then present our idea of combining *heuristic algorithms* with scaling down schemes. Some innovative schemes are suggested to maximize a defined *satisfaction measure* and at the same time support the allocation of the *assured bandwidth* for each subscriber.

In Section 2, the system's quality of service parameters is presented and performance criteria are defined. The resource management process is presented in Section 3, including the mathematical background for the MKP problem with extension, together with a problem reformulation that may suggest a *scaling down* strategy as a natural way to solve the extended MKP problem.

Section 4 focuses on the algorithms used for the resource allocation, while Section 5 shows the significance of the scal-ing schemes. Several schemes are presented, based on the quality of service parameters. Then a simplified example is presented to illustrate the whole process. The techniques used to model the system dynamics are presented in Section 6, while Section 7 compares the performances obtained in simulation. Finally, we draw our conclusions in Section 8.

2. System Parameters and Performance Criteria

2.1. The Quality of Service Parameters. Parameters of quality of service determine the capability of providing better service to network traffic. The system we study assumes some quality of service characteristics for each subscriber. The algorithms presented here take into consideration two such parameters.

(a) Priority Level. A priority factor w_k is assigned to each subscriber type, such that higher level subscribers get priority in allocation of bandwidth following user request.

(b) Assured Level. Subscribers are guaranteed (with a high probability) to receive the bandwidth they request up to some assured amount Ω_k. Any bandwidth above this is optional, up to a maximal value determined by PIR.

These two parameters are used in the definition of the *satisfaction measure* function and play an important role in algorithm design and optimization of bandwidth utilization.

2.2. Disconnections. The fact that the DVB-S2 links were designed for fixed center frequency and symbol rate would cause disconnections while changing them. Therefore, allocating a new bandwidth to a user causes a disconnection to the subscriber. To take into account this fact, we use *effective allocated bandwidth* A related to the *allocated bandwidth* A_0 according to:

$$A = \begin{cases} \beta \cdot A_0, & \text{(if disconnected)}, \\ A_0, & \text{(otherwise)}, \end{cases} \tag{1}$$

where $\beta < 1$ is a constant expressing the effect of disconnection period, and the assumed dissatisfaction it causes. In this paper, we narrow ourselves to the case where users are not moved in the frequency range if they have no request for a change.

2.3. The Satisfaction Criterion. Resource allocation is conducted according to the requests of subscribers, their priority levels, and their assured bandwidths. For a subscriber with assured bandwidth Ω, a requested bandwidth R, and an effective allocated bandwidth A, and assuming that $R \geq A$, we suggest the satisfaction measure S that may be calculated for a factor $\alpha < 1$:

$$S(R, A, \Omega) = \begin{cases} \dfrac{\Omega + \alpha \cdot (A - \Omega)}{\Omega + \alpha \cdot (R - \Omega)}, & (R > A \geq \Omega), \\ \dfrac{A}{\Omega + \alpha \cdot (R - \Omega)}, & (R > \Omega \geq A), \\ \dfrac{A}{R}, & (\Omega > R \geq A). \end{cases} \tag{2}$$

The satisfaction function ranges from 0 to 1. The maximum satisfaction value $S = 1$ is reached when the allocated bandwidth is equal to the requested bandwidth. Also, one can see that when $A > \Omega$, the contribution of any increment ΔA to the function is smaller by a factor of $\alpha < 1$ than the contribution of the same increment ΔA when $A < \Omega$.

In order to reduce the number of variables involved in this formula, we express the satisfaction function in terms of the *normalized bandwidth variables*:

$$r = \frac{R}{\Omega}, \qquad a = \frac{A}{\Omega}, \tag{3}$$

such that the satisfaction measure may be defined as:

$$S(r, a) = \begin{cases} \dfrac{1 + \alpha \cdot (a - 1)}{1 + \alpha \cdot (r - 1)}, & (r > a \geq 1), \\ \dfrac{1}{1 + \alpha \cdot (r - 1)}, & (r > 1 \geq a), \\ \dfrac{a}{r}, & (1 > r \geq a). \end{cases} \tag{4}$$

This is a concave function increasing monotonically both in the r direction and in the a direction.

For a given set of users with normalized requests and allocated bandwidths r_i, a_i and priority factors w_i, the total (weighted) satisfaction $\overline{S}$ is defined as:

$$\overline{S} = \frac{\sum_i w_i \cdot S(r_i, a_i)}{\sum_i w_i}. \tag{5}$$

The total satisfaction function gets the maximum of 1 when all requests are fulfilled. This quantitative measure may be used to compare the performance of the various algorithms and schemes.

2.4. The Requests Scale Down Criterion. While resource allocation algorithms fit the bandwidth requests into the *holes* in the spectrum, some *holes* may become *under populated* and some *over-populated*.

The first situation (*under population*) happens when the sum of requests inserted into a *hole* does not fully cover it.

The other one (*over population*) happens when the sum of requests assigned into a *hole* is greater than the capacity of the *hole*. In this situation the requests assigned to that *hole* should be scaled down to fit into the *hole* capacity.

Let $\{R_j^{(k)}\}$ be the set of requests allocated into a *hole* $H^{(k)}$, Then the total amount δ of the requests bandwidth that is scaled down is:

$$\delta = \sum_k \max\left(\left[\sum_j R_j^{(k)}\right] - H^{(k)}, 0\right). \tag{6}$$

The amount δ is the *requests scale down* factor. It measures the effectiveness of the algorithms in finding a good arrangement of the requests into the gaps (*holes*) in spectrum. A lower value indicates better algorithm performance.

3. The Resource Allocation Process

The disconnections caused while changing frequency and symbol rates pose a serious constraint in the design of optimal allocation algorithms. Therefore, links of terminals that did not require any change should stay in the same location in the spectrum, while new requests should be allocated only in the free portions (*holes*) of the spectrum.

In this part, we will show that this problem is equivalent to the multi-knapsack problem (MKP) [6] known in computational complexity theory and then draw a scheme for a computational process to cope with it.

3.1. Resource Allocation as a Complexity Theory Problem. The connection between the MKP problem and satellite communication resource allocation was already reported by Birmani [7]. However, his work deals with *power allocation* and burst scheduling problems, which are *inherently quite different* in their nature than the *bandwidth allocation* problems in our concern.

The association between resource allocation in communication and the MKP problem was already observed by Rajkumar et al. [8]. Their research was on a QoS-based Resource Allocation Model (QRAM) for satisfying multiple quality-of-service dimensions in a resource-constrained environment. Using this model, available system resources can be apportioned across multiple applications, thus maximizing the net utility that accrues to the end users of those applications. They showed that the Q-RAM problem of finding the optimal resource allocation to satisfy multiple QoS dimensions is NP hard. There is a similarity between the resources, QoS, and utility factor in their research and the bandwidth, priority level, and satisfaction function in this research. However, in our work the satisfaction function is different than their utility factor and the methods we suggest are simpler to implement.

The task of the resource allocation algorithms is to find an optimal match between a set of N requests $\{R_i \mid 0 \le i < N\}$ and a set of M holes $\{H^{(j)} \mid 0 \le j < M\}$ in a spectrum such that *all* the N_j requests are inserted into those holes:

$$\sum_{i=0}^{N_j} R_{K_j(i)} \quad \le \quad H^{(j)}, \tag{7}$$

where $\{K_j\}$ are M disjoint subsets of indexes: the size of set K_j is N_j, such that their union covers the set $\{1, \ldots, N\}$:

$$\bigcap K_j = \phi, \qquad \bigcup K_j = \{i \mid i \le N\}, \tag{8}$$

where $K_j(i)$ is the ith element in the set K_j.

The algorithm theory classifies such a problem as a variation of the *multi-knapsack problem.* Given M containers (knapsacks) with capacities $H^{(j)}$ and N items with values R_i, the task is to split the set $\{R_i\}$ into M disjoint subsets, such that the sum of item values in each subset will not exceed the given capacities $H^{(j)}$ as in (7).

Although the set of inequalities in the problem definition resembles a typical linear programming formulation, it is not a linear programming problem. Instead, it belongs to a family of *subset sum* problems known as knapsack problems.

The knapsack problem is part of the family of combinatorial NP-complete problems [9], meaning that it is computationally difficult to solve in general (except for a $O(M^N)$ brute force solution). It is difficult enough to be chosen as a trapdoor cryptographic function (i.e., the Merkle-Hellman cryptosystem).

Yet the resource allocation problem does not fit the exact definition of a knapsack problem. There is one essential difference: when (7) cannot be fully satisfied, then the target of optimization should be replaced by:

$$\sum_{i=0}^{N_j} C_i^{(j)} \cdot R_{K_j\,(i)} \le H^{(j)}, \quad \left(0 < C_i^{(j)} \le 1\right), \tag{9}$$

such that the factors $C_i^{(j)}$ are maximal.

This is a "softer" version of the original problem (7). It extends the problem adding some degree of complexity (finding the factors $C_i^{(j)}$). On the other hand, it eases the task somewhat, adding several degrees of freedom so that a solution can be found, even in situations where the original problem is practically insolvable.

So, because this is an NP-Hard problem, we shall not try to solve it fully. Instead we may design some practical heuristic algorithms and reach suboptimal solutions [10].

3.2. The Resource Management Process. The resource allocation problem is too complex to be solved at once. Instead of solving (9) directly by a single algorithm, we suggest splitting it into three subproblems:

(1) find an initial approximation to the $C_i^{(j)}$ factors;

(2) solve the main resource allocation problem;

(3) find the final values of the $C_i^{(j)}$ factors.

Following this model, the entire resource management process consists of three stages, as illustrated in Figure 1.

(1) The prescaling Step. In order for the resource allocation algorithms to operate properly, subscriber requests for bandwidth *cannot* exceed the available bandwidth budget. However, this situation does occur frequently. To cope with this problem, we suggest the *pre-scaling* process, to scale down subscriber requests according to the priority factors w_i and the assured bandwidths Ω_j.

(2) The resource allocation algorithm finds the best fit between the requests and the *holes* in the spectrum.

(3) The postscaling step is performed in one of the following situations:

(a) if the sum of the allocated requests assigned to a *hole* is bigger than the size of the *hole*, then the requests should be *downscaled*;

(b) if the sum of the allocated requests assigned to a hole is smaller than the size of the *hole*, and the requests were downscaled (by means of pre-scaling stage), then they may be *upscaled*. The allocated bandwidths are scaled up such that their new value will not exceed the original requests, and in addition their sum will not exceed the size of the hole.

In either case, the scaling is conducted according to the priority factors w_i and the assured bandwidths Ω_i in a manner very similar to the prescale stage.

The feedback in Figure 1 marks the transition to the next phase of the algorithm, when new requests are generated.

4. The Resource Allocation Algorithms

Two heuristic algorithms are presented to solve the main resource allocation problem:

(i) the Fast MKP solver;

(ii) the IBF (Insert to Best Fit) MKP solver.

Both algorithms are classified as *greedy algorithms*. At each stage they take the locally optimal choice rather than aiming for the global optimum. Such algorithms are widely used to solve heuristically *dynamic programming* problems.

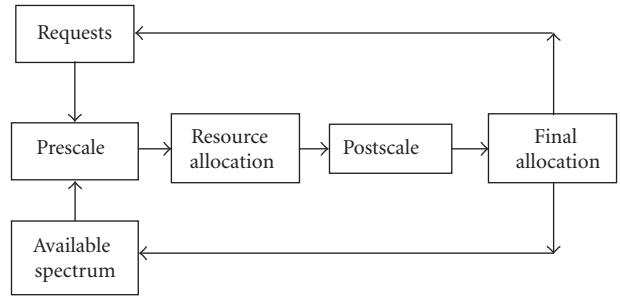


Figure 1: The resource allocation process flow chart.

4.1. The Fast MKP Algorithm. When allocating a request to a hole, we define the residue remainder after the allocation, that is, the size of the hole less the size of the request. The *Fast MKP* algorithm adopts a strategy of *largest residues first*, that is, it fits the largest requests to *holes* with the largest residue in decreasing order.

In order to perform this efficiently, we maintain a binary tree of *hole* items (sorted by the residue size) to enable a *search and insert* operation in $O(\log(M))$ time steps [11].

The algorithm consists of the following steps.

(1) Make a sorted list of requests in *decreasing order* (largest to smallest) and a binary tree of residues associated with the *hole* items.

(2) Fit the *largest request* to the biggest residue *hole* item.

(3) Update the residue value corresponding to the *hole* item and insert it back into the tree according to the residue value.

(4) Discard the request from the list of requests.

(5) Repeat steps (2)–(4) until the request list is empty.

This algorithm is efficient with respect to run time and memory consumption.

4.2. The IBF MKP Algorithm. This algorithm adopts a strategy of *best fit residue*, that is, it fits requests in decreasing order into *holes* with the residue that fits best (if no residue fits, it inserts to the largest one—). A *best fit* for any R_j over a *decreasing* series of residues $\{Z_i\}$, is any Z_i, such that $Z_{i+1} < R_j \leq Z_i$.

(1) Start with a sorted list of requests in *decreasing order* (largest to smallest) and a binary tree of residues associated with the *hole* items.

(2) Insert the largest request into the *hole* item, with the smallest residue that is larger than the request.

(3) If there is no such residue, insert the request into the *hole* with the largest residue.

(4) Update the residue value corresponding to the *hole* item and insert it back into the tree according to the residue new value.

(5) Discard the request from the list of requests.

(6) Repeat steps (2)–(5) until the request list is empty.

The purpose of stage (3) in the algorithm is to ensure that *all* the requests are inserted into the *holes*, even when *holes* become *overpopulated*. This may involve temporarily assigning negative values to some residues. This is quite legitimate, because the residue values have instrumental meaning only. The *postscaling* stage described in next section would resolve this problem.

5. The Scaling Schemes

The algorithms presented above perform poorly under a shortage of bandwidth resources. We suggest activating some scaling schemes when the sum of the requests is greater than the sum of the holes. In those cases, such schemes can share *fairly* the available bandwidth resources, taking into consideration the *quality of service* parameters in order to resolve the users' diversity issues.

For a set of requests $\{R_i\}$ and a set of *holes* $\{H^{(j)}\}$ such that $R = \sum R_i$ and $H = \sum H_i$ we introduce *scaling schemes* to take place when $R > H$.

Introducing scaling stages into the allocation process is crucial. It is analogous to *adding filters* at the entrance and outlet of an electronic system in order to adjust the signal to the dynamic range of a system. The design of scaling schemes may dramatically influence the overall performance of the allocating process just like filters that affect system's behavior.

We present first the schemes for the *downscaling* performed in the prescaling step. In correspondence with the quality of service levels [12], we consider several levels of scaling schemes in accordance with the number of parameters taken into consideration:

(i) the basic scheme, which takes no parameters;

(ii) *the priority-oriented scheme*, which takes into consideration the priority factors w_i only;

(iii) advanced schemes, taking into consideration the assured bandwidth Ω_i as well as the priority factors w_i.

5.1. The Basic Scaling Scheme. In absence of any priority criteria for sharing the bandwidth, the requests are simply scaled down proportionally to their value:

$$R_k^{(\text{new})} = \frac{H}{R} \cdot R_k. \tag{10}$$

Such a scheme corresponds to a network system that supports *best-effort* subscribers only, with no quality of service parameters.

5.2. The Priority-Oriented Scaling Scheme. This scheme takes into consideration only the *priority factors* w_i. Such a scheme is applicable for network systems that support *at least* the *differentiated service* level.

The amount of bandwidth to be reduced is the difference between the sum of requests and the sum of holes:

$$D = R - H. \tag{11}$$

This amount should be shared proportionally to the requests R_j (larger requests are reduced more than small ones) and inversely proportional to the priority factors w_j (higher priority requests are reduced less than the lower ones). So, the request R_j should be reduced by amount d_j given by: $d_j = C \cdot (R_j/w_j)$ such that $D = \sum_j d_j$ is the total reduction.

Hence: $D = C \cdot (\sum_j R_j/w_j)$ where C is some proportion factor that can be written as:

$$C = \frac{D}{\sum_k R_k/w_k} = \frac{R - H}{\sum_k R_k/w_k}. \tag{12}$$

And finally, the requests are downscaled according to:

$$R_j^{(\text{new})} = \left(1 - \frac{C}{w_j}\right) \cdot R_j. \tag{13}$$

This scheme works unconditionally (but it does not take into consideration the assured bandwidth Ω).

5.3. Advanced Schemes. High-quality network systems should take into consideration the subscribers' *assured bandwidths* Ω_i, as well as their *priority factors* w_i. The constraint imposed by the assured bandwidth condition can be expressed as:

$$R_i^{(\text{new})} \geq \min(R_i, \Omega_i). \tag{14}$$

This condition could be met only when:

$$\sum H_i \geq \sum \min(R_i, \Omega_i). \tag{15}$$

When this condition is satisfied, we can allocate bandwidth taking into account both the priority factors and the assured bandwidths. Otherwise, the *priority-oriented scaling scheme* should be used.

Our strategy is to grant all the requests below the assured bandwidth and scale down *only* the others.

The two schemes we present here have the same "prolog", and differ only in the final stage.

(0) Start with a set of indexes $I = \{1, \ldots, N\}$, where N is the number of requests.

(1) Split I into 2 subsets:

$$I^{(-)} = \{i \mid R_i \leq \Omega_i\} \text{ and}$$
$$I^{(+)} = \{i \mid R_i \leq \Omega_i\} \text{ (the complementary set).}$$

(2) Then define

$$H^{(+)} = H - \sum_{i \in I^{(+)}} R_i, \qquad \Omega^{(+)} = \sum_{i \in I^{(+)}} \Omega_i. \tag{16}$$

(3) $R_i^{(\text{new})} = R_i,\ i \in I^{(-)}$ ("grant requests less than user's assured bandwidth").

(4) Scale the requests in $I^{(+)}$ using one of the schemes described next.

(i) Difference-Oriented Scaling. Using the *difference* $(R_i - \Omega_i)$ weighted by the priority factors w_i, we get:

$$R_i^{(\text{new})} = \Omega_i + C \cdot w_i \cdot (R_i - \Omega_i), \quad i \in I^{(+)}. \tag{17}$$

In similar arguments used in developing the *priority-oriented scaling scheme*, we find:

$$C = \frac{H^{(+)} - \Omega^{(+)}}{\sum_{i \in I^{(+)}} (R_i - \Omega_i) \cdot w_i}. \tag{18}$$

(Note that the denominator $(R_i - \Omega_i) \cdot w_i$ is always positive because $R_i > \Omega_i$ for all $i \in I^{(+)}$).

(ii) Ratio-Oriented Scaling. Using the *ratio* (R_i/Ω_i) weighted by the priority factors w_i, we get:

$$R_i^{(\text{new})} = \Omega_i + C \cdot w_i \cdot \left(\frac{R_i}{\Omega_i}\right), \quad i \in I^{(+)}, \tag{19}$$

where again by similar arguments we get:

$$C = \frac{H^{(+)} - \Omega^{(+)}}{\sum_{i \in I^{(+)}} (R_i/\Omega_i) \cdot w_i}. \tag{20}$$

After the prescaling procedure, the algorithms in Section 4 are applied. This prescaling is a necessary but not sufficient condition to be able to complete the procedure. However, the postscaling can be used to complete it.

5.4. The Postscaling Schemes. The postscaling stage makes the final correction to the allocated bandwidths. It may involve more downscaling to some of the requests. This happens when the allocating algorithm inserts to a hole a set of requests whose sum exceeds the hole size and assigns a negative value to the residue associated with the hole. In this case, the downscaling is performed in essentially same way as in the prescale step, except that the calculations are made separately for each *hole* j. Hence H_j is used instead of H and $\{R_i^{(j)}\}$, and the set of requests allocated to H_j would replace R_i.

There are situations when the allocating algorithm assigns to a hole requests whose sum is less than the *hole* size. In this case, if some of the requests were over-decreased in the prescaling stage, they should be now increased back (up to the original amount) to fit the hole size.

5.5. Concluding Remarks. We have presented here some scaling schemes to support *fair* sharing of available bandwidth resources in shortage conditions.

Naturally, the *advanced schemes* are expected to perform better than the *priority oriented scheme* because they take into consideration the assured bandwidths as well as the priority factors.

However, as we have pointed out, the other scaling schemes should not be abandoned. In some exceptional situations (heavy traffic periods for instance), when condition (15) is not satisfied, the *priority oriented* scheme (or the *Basic* one) should be used instead.

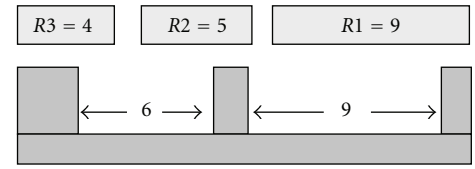


Figure 2: The initial state of *holes* and requests.

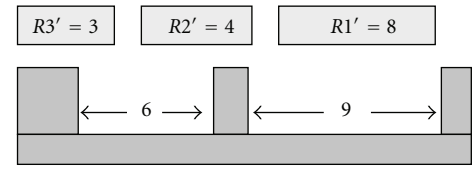


Figure 3: The state of *holes* and requests *after prescaling*.

5.6. An Illustrative Example for the Allocation Cycle. To illustrate the whole resource allocation cycle, consider the following simplified example. Consider three requests issued to the system:

$$R_1 = 9, \qquad R_2 = 5, \qquad R_3 = 4. \tag{21}$$

When there are two available "holes" in the spectrum:

$$H_1 = 9, \qquad H_2 = 6. \tag{22}$$

As shown in Figure 2.

The sum of the requests $\sum R_j = 18$ is greater than the available space in the spectrum $\sum H_i = 15$, meaning there is deviation of 3, and a scaling phase should take place to address it.

(i) Prescaling. For the sake of simplicity and in order to make it intuitive to the reader, tedious calculations involving QoS parameters considerations are dropped. Instead, we will focus on illustrating the concepts of the scaling stages.

To solve the deviation problem, the prescaling phase can reduce each request by 1. For instance,

$$R = \{9, 5, 4\} \longrightarrow \{8, 4, 3\}. \tag{23}$$

As shown in Figure 3.

This way $\sum R_j = 15$, which is exactly the amount of available space.

(ii) Allocation. The *residue* of H_i was denoted as Z_i. Initially set: $\{D\} = \{H\}$. Then:

(1) insert $R_1 = 8$ into H_1 (with $Z_1 = 9$) such that the residue is updated to $D_1 = 1$;

(2) insert $R_2 = 4$ into H_2 (with $Z_2 = 6$) which makes this residue become $Z_2 = 2$;

(3) finally, insert $R_3 = 3$ into H_2 (with $Z_2 = 2$). This makes the *residue* Z_2 receive a *negative value*: $Z_2 = -1$!

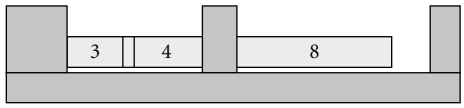


Figure 4: The requests inserted to the *holes by the allocation algorithm*. Notice the *overlapping* between R'_3 and R'_4.

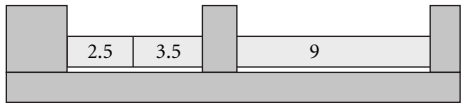


Figure 5: The final state *after the postscaling*.

This state of affairs is presented in Figure 4. The overpopulation situation in H_2 caused by the allocation algorithm is expressed by the *overlapping* R_2 and R_3 in the figure.

As stated in Section 4.2, assigning a negative value to some residues is legitimate as an intermediate stage in calculation, and the task of the *postscaling* phase is to fix this situation.

(iii) Postscaling. This phase fine tunes the bandwidths allocated. Recall that originally the request R_1 was 9, and then it was reduced to 8 and allocated to H_1, which has a capacity of 9. Obviously the original request can be granted. Thus the final allocation for R_1 should be $A_1 = 9$.

R_2 and R_3 were allocated to H_2 which has a capacity of 6, but $R_2 + R_3 = 7$, so they should be reduced again to: $A_2 = 3.5$ and $A_3 = 2.5$ such that $A_2 + A_3 = 6$ to fit the $H_2 = 6$ capacity.

So the final resource allocation yields:

$A_1 = 9$ inserted into $H_1 = 9$ and $\{A_2 = 3.5,\ A_3 = 2.5\}$ inserted into $H_2 = 6$. (See Figure 5).

6. Modeling the Resource Allocation Process

In order to simulate the system dynamics under the resource allocation process described above, we derived a model to produce the network traffic and the inputs to the process. We distinguish between the *demands* subscribers have and the *requests* they issue to the system. New demands for bandwidth emerge for each subscriber in random time steps. We assume that the demands ν_k of subscriber k are *gamma* distributed.

$$\nu_k \sim \Gamma(m, \lambda_k). \tag{24}$$

The gamma distribution [13] is a two-parameter family of continuous probability distributions. It has a shape parameter m, a scale parameter λ_k, and the average value:

$$E[\nu_k] = m \cdot \lambda_k. \tag{25}$$

The traffic models based on random variables with Gamma distribution are used in applications such as video and speech [14, 15]. Such a random variable has a probability distribution with a shape similar to a Gaussian random variable but has only positive values.

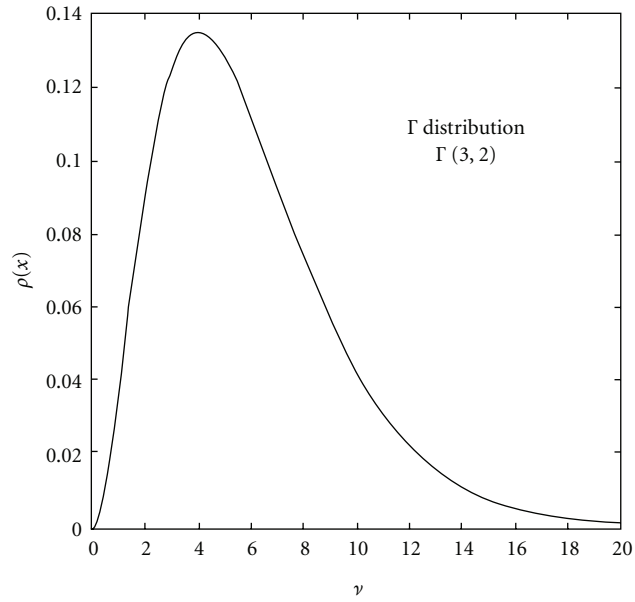


Figure 6: Gamma distribution: Γ(3,2).

The parameters are chosen such that the mean demand of subscriber k, ν_k, is proportional to Ω_k, the assured bandwidth of subscriber k, by a constant factor μ:

$$E[\nu_k] = \mu \cdot \Omega_k. \tag{26}$$

So comparing (25) and (26):

$$\lambda_k = \frac{\mu \cdot \Omega_k}{m}. \tag{27}$$

In the simulation, the values used are: $\mu = 2.0$ and $m = 3$, as shown in Figure 6.

We may assign a probability p for new demand to emerge such that in time step n we have demand $\nu_k^{(n)}$:

$$\nu_k^{(n)} = \begin{cases} \nu \sim \Gamma(m, \lambda_k), & \text{with probability } p, \\ 0, & \text{with probability } (1 - p). \end{cases} \tag{28}$$

We assume that *very often user requests are not fully granted*. In this situation, a subscriber should request the remainder $R_k - A_k$ in the next time step, added to a new demand (if one emerges). This can be modeled by assigning a variable Q_k to each subscriber k to accumulate requests R_k, subtract the allocated bandwidth A_k, and add demands ν_k:

$$Q_k^{(n+1)} = Q_k^{(n)} + R_k^{(n)} - A_k^{(n)} + \nu_k^{(n+1)}. \tag{29}$$

Stability considerations suggest limiting the request amount a subscriber k may issue. In Section 1.1, we called this limit the Peak Information Rate (PIR). We set this limit to $\kappa \cdot \Omega_k$, where $\kappa > 1$ is a constant ($\kappa = 2$ in our simulation):

$$R_k^{(n+1)} = \min\left(Q_k^{(n+1)}, \kappa \cdot \Omega_k\right). \tag{30}$$

Table 1: The simulation configuration.

Parameter		Equation	Value
Satisfaction coefficient	α	(2)	0.75
Disconnection penalty factor	β	(1)	0.9
Mean demand to assured bandwidth ratio	μ	(18)	2.0
Demand update rate	p	(21)	0.25
Threshold for new request	γ	(24)	0.40
Request bound to assured bandwidth ratio	κ	(23)	2.0

Table 2: Comparison of the "requests scale-down" between the algorithms.

	Fast-MKP	IBF-MKP
$\delta = 0$	34.0%	43.2%
$0 < \delta \leq 250$	6.0%	9.8%
$250 < \delta \leq 500$	5.4%	8.8%
$500 < \delta \leq 1000$	11.0%	14.6%
$1000 < \delta \leq 2000$	19.8%	22.6%
$2000 < \delta \leq 4000$	23.0%	35.2%
$\delta > 4000$	6.0%	10.0%

Table 3: Comparison of the *requests scale-down* for different scaling schemes.

	Priority oriented	Difference oriented	Ratio oriented
$\delta = 0$	32.0%	44.6%	48.2%
$0 < \delta \leq 25$	5.8%	8.4%	7.2%
$25 < \delta \leq 50$	8.2%	11.4%	6.4%
$50 < \delta \leq 100$	10.4%	20.6%	8.0%
$100 < \delta \leq 200$	14.4%	24.2%	12.2%
$200 < \delta \leq 400$	17.8%	20.4%	13.8%
$\delta > 400$	11.4%	15.0%	7.8%

Finally, in order to avoid the penalty of unwanted disconnections associated with new bandwidth allocations, a restriction is superimposed on issuing new requests. Only changes exceeding some threshold can justify issuing new requests. To assure this, we set up a rule: *a new request shall be issued only when:*

$$\left| R_k^{(n+1)} - A_k^{(n)} \right| \geq \gamma \cdot A_k^{(n)}. \tag{31}$$

In the simulation, $\gamma = 0.4$ is used.

7. Comparison between Algorithms

Algorithms can be ranked either by their efficiency or by the quality of the results they produce. We shall examine and compare our algorithms in the following aspects:

(i) algorithm complexity (memory and time consumption);

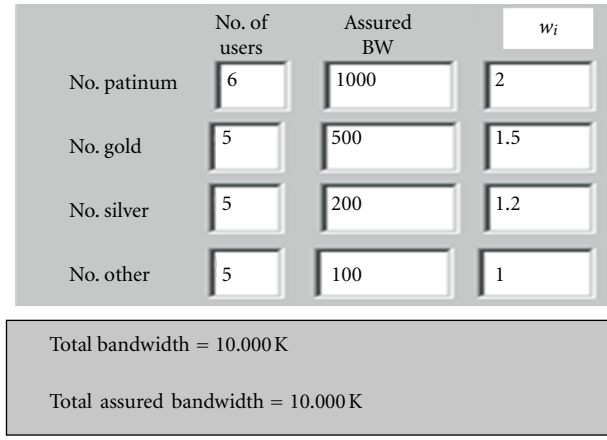


Figure 7: The system configuration in a simulation snapshot.

(ii) distribution of the satisfaction measure S;

(iii) distribution of the requests scale down δ.

We considered a typical system with 4 types of subscribers. Each type has its priority factor w_k and assured bandwidth Ω_k. The system setup is shown in a simulation snapshot presented in Figure 7.

The other simulation parameters are presented in Table 1.

7.1. The Complexity of the Algorithms. Let us consider the complexity of the algorithms (without the scaling part) for N new requests and M *holes* in the spectrum.

The time complexity of both Fast-MKP and IBF-MKP algorithm is bounded by $O(N \cdot \log(M))$ *integer operations*, as the *search and insert* operation in a binary tree is $O(\log(M))$.

The memory complexity of both algorithms is $O(N+M)$. However, we observed (experimentally) that the IBF-MKP algorithm runs slower than Fast-MKP by no more than 20% on the average. So, both algorithms are very efficient in runtime and memory consumption.

7.2. The Distributions of the Satisfaction Measure. The distributions for the satisfaction measure for the IBF-MKP and the Fast-MKP algorithms are presented by the histograms in Figures 8 and 9, obtained by simulation without the effect of the scaling schemes. For the criterion of satisfaction, the IBF-MKP algorithm achieves better results.

The effect of the scaling schemes on the satisfaction measure is shown in Figures 10, 11, and 12. The results are presented for the IBF-MKP algorithm only. The scaling schemes do improve results over those obtained without the scaling. The *ratio-oriented* scaling scheme yields better results than *Difference Oriented*, while the *Priority Oriented* scaling scheme is in third place.

7.3. The Distributions of the "Requests Scale-Down". The simulation records the amount of bandwidth scaled down in the resource allocation. Table 2 compares the distribution

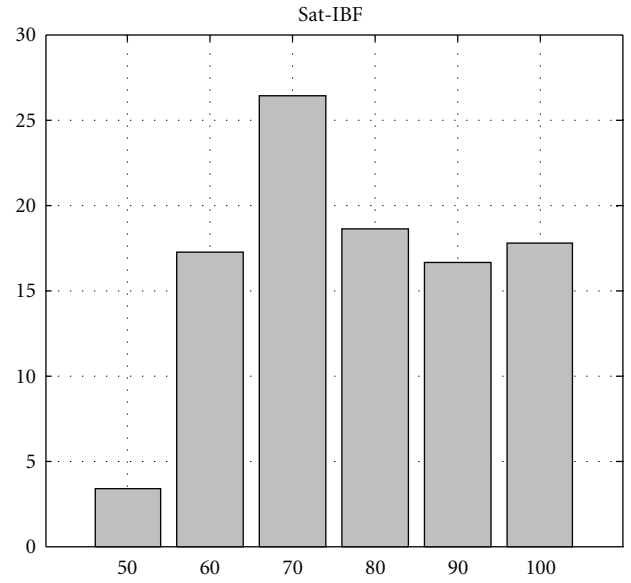


Figure 8: Satisfaction histogram IBF-MKP algorithm with *priority-oriented* scaling scheme.

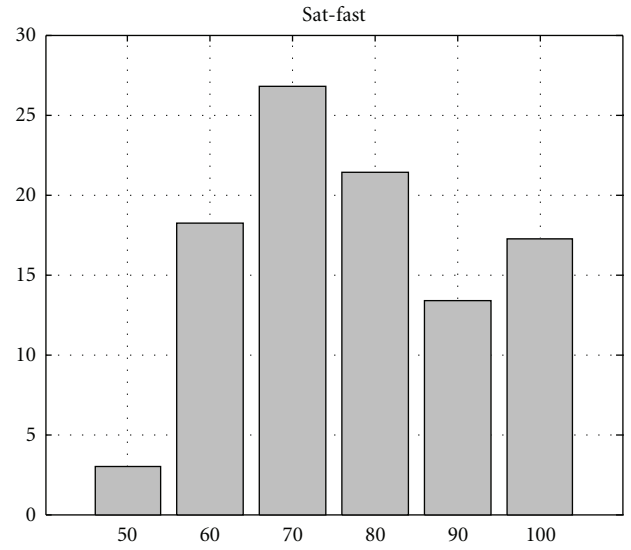


Figure 9: Satisfaction histogram for Fast-MKP Algorithm with priority-oriented scaling scheme.

of this variable for the two algorithms (without utilizing the scaling schemes).

Clearly, the IBF-MKP algorithm achieves better results for the average amount of requests scale-down in the allocation process.

The effect of scaling schemes on the *requests scale-down* factor δ is compared in Table 3, which presents results where again the algorithm is the IBF-MKP.

As expected, scaling schemes improve the *scale-down* results. The *ratio-oriented* gives the best results, while the *difference-oriented* is somewhat less effective, and the *priority-oriented* offers the least effectiveness.

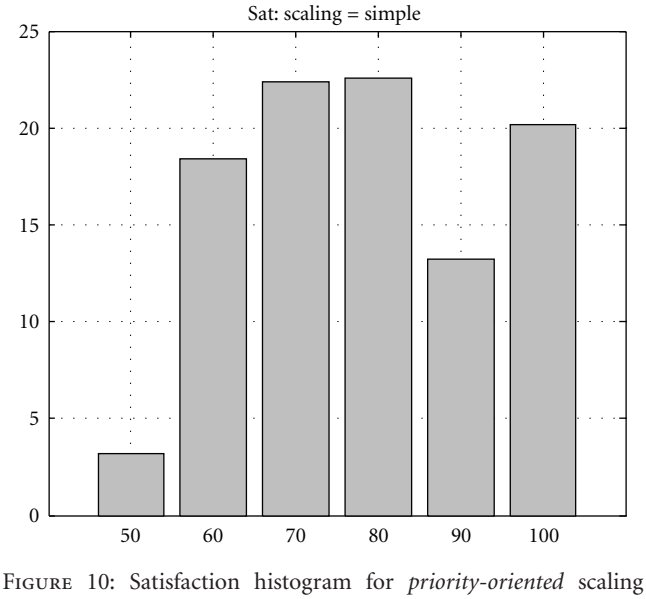


Figure 10: Satisfaction histogram for *priority-oriented* scaling scheme.

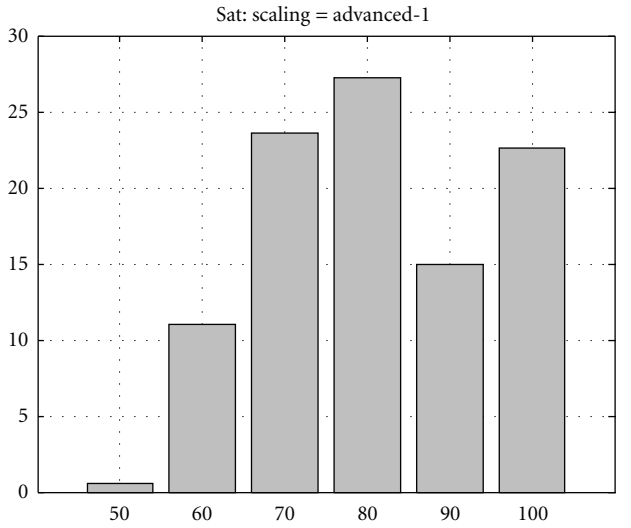


Figure 11: Satisfaction histogram for *difference-oriented* scaling scheme.

The difference is probably due to the ability to grant all requests below the assured bandwidth. This approach makes these advanced scaling schemes superior to other schemes.

8. Conclusion

In a rapidly evolving communications market with ever-growing demand for more bandwidth, a good resource management mechanism is essential for efficient digital communication of any kind. The algorithms suggested here together with the scaling schemes successfully perform this task with reasonable computational complexity, producing a high-quality resource allocation with remarkable performance.

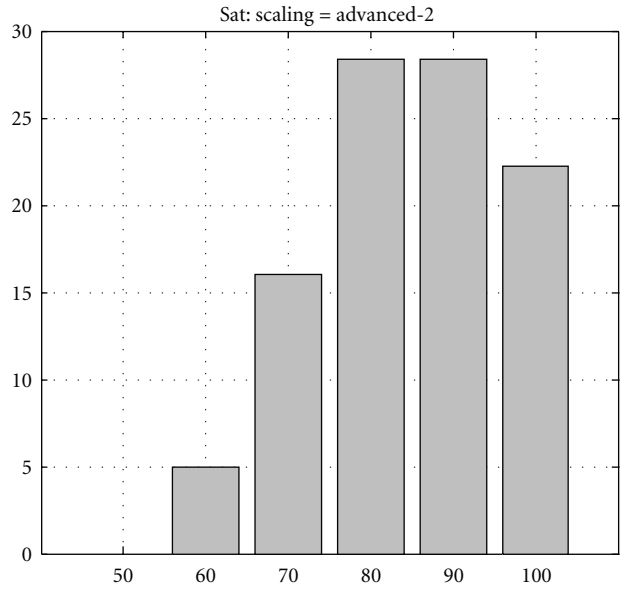


Figure 12: Satisfaction histogram for *ratio-oriented* scaling scheme.

Acknowledgments

The authors wish to acknowledge the cooperation of Ayecka Communication Systems Ltd. and the support of the MAGNETON program of the Israel Ministry of Industry, Trade and Labor.

References

[1] ETSI: EN 302 307, Digital Video Broadcasting (DVB), "Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)," *European Standard* (Telecommunications series), 2005.

[2] ETSI: EN 300 421, Digital Video Broadcasting (DVB), "Framing structure, channel coding and modulation for 11/12 GHz satellite services (DVB-S)," *European Standard* (Telecommunications series),1994.

[3] ETSI: EN 301 790, Digital Video Broadcasting (DVB), "Interaction channel for satellite distribution systems (DVB-RCS)," *European Standard* (Telecommunications series), 2005.

[4] G. Giambene, *Resource Management in Satellite Networks Optimization and Cross-Layer Design*, Springer, 2007.

[5] S. I. Wayer and A. Reichman, "Resource management in satellite communication systems—Heuristic algorithms," in *Proceedings of the IEEE 26th Convention of Electrical and Electronics Engineers in Israel (IEEEI '10)*, pp. 342–346, November 2010.

[6] D. Pisinger, "A Minimal Algorithm for the Bounded Knapsack Problem," *INFORMS Journal on Computing*, vol. 12, no. 1, pp. 75–82, 2000.

[7] V. Birmani, *Resource allocation for Ka-band broadband satellite systems [M.S. thesis]*, University of Maryland, 1999.

[8] R. Rajkumar, C. Lee, J. P. Lehoczky, and D. P. Siewiorek, "Practical solutions for QoS-based resource allocation problems," in *Proceedings of the 19th IEEE Real-Time Systems Symposium*, pp. 296–306, December 1998.

[9] S. Goddard, "P, NP, and NP-Complete," CSCE 310J lecture notes on "Data Structures & Algorithms", University of Nebraska-Lincoln, http://www.cse.unl.edu/~goddard/Courses/CSCE310J/Lectures/Lecture10-NPcomplete.pdf.

[10] M. Hifi, M. Michrafy, and A. Sbihi, "Heuristic algorithms for the multiple-choice multidimensional knapsack problem," *Journal of the Operational Research Society*, vol. 55, no. 12, pp. 1323–1332, 2004.

[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, 2nd edition, 2001.

[12] Quality of Service Networking, http://docwiki.cisco.com/wiki/Quality_of_Service_Networking.

[13] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, NY, USA, 2nd edition, 1984.

[14] M. Menth, A. Binzenhöfer, and S. Mühleck, "Source models for speech traffic revisited," *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1042–1051, 2009.

[15] O. Rose, "Simple and efficient models for variable bit rate MPEG video traffic," *Performance Evaluation*, vol. 30, no. 1-2, pp. 69–85, 1997.

Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2012, Article ID 840513, 16 pages
doi:10.1155/2012/840513

Research Article

Robustness Maximization of Parallel Multichannel Systems

Jean-Yves Baudais,[1] Fahad Syed Muhammad,[2] and Jean-François Hélard[2]

[1] *National Center for Scientific Research (CNRS), The Institute of Electronics and Telecommunications of Rennes (IETR), UMR 6164, 35708 Rennes, France*
[2] *Université Européenne de Bretagne, INSA, IETR, UMR 6164, 35708 Rennes, France*

Correspondence should be addressed to Jean-Yves Baudais, jean-yves.baudais@insa-rennes.fr

Received 27 February 2012; Accepted 10 May 2012

Academic Editor: Shuo Guo

Bit error rate (BER) minimization and SNR-gap maximization, two robustness optimization problems, are solved, under average power and bitrate constraints, according to the waterfilling policy. Under peak power constraint the solutions differ and this paper gives bit-loading solutions of both robustness optimization problems over independent parallel channels. The study is based on analytical approach, using generalized Lagrangian relaxation tool, and on greedy-type algorithm approach. Tight BER expressions are used for square and rectangular quadrature amplitude modulations. Integer bit solution of analytical continuous bitrates is performed with a new generalized secant method. The asymptotic convergence of both robustness optimizations is proved for both analytical and algorithmic approaches. We also prove that, in the conventional margin maximization problem, the equivalence between SNR-gap maximization and power minimization does not hold with peak-power limitation. Based on a defined dissimilarity measure, bit-loading solutions are compared over Rayleigh fading channel for multicarrier systems. Simulation results confirm the asymptotic convergence of both resource allocation policies. In nonasymptotic regime the resource allocation policies can be interchanged depending on the robustness measure and on the operating point of the communication system. The low computational effort leads to a good trade-off between performance and complexity.

1. Introduction

In transmitter design, a problem often encountered is resource allocation among multiple independent parallel channels. The resource can be the power, the bits or the data, and the number of channels. The resource allocation policies are performed under constraints and assumptions, and the independent parallel channels can be encountered in multitone transmission.

Independent parallel channels result from orthogonal design applied in time, frequency, or spatial domains [1]. They can either be obtained naturally or in a situation where the transmit and receive strategies are to orthogonalize multiple waveforms. The orthogonal design can also be applied in many communication scenarios when there are multiple transmit and receive dimensions. Orthogonal frequency-division multiplexing (OFDM) and digital multitone (DMT) are two successful commercial applications for wireless and wireline communications with orthogonality in the frequency domain.

To perform resource allocation, relations between various resources are needed, and one is the channel capacity. This capacity of n-independent parallel Gaussian channels is the well-known sum of the capacity of each channel

$$\mathcal{C} = \sum_{i=1}^{n} \mathcal{C}_i = \sum_{i=1}^{n} \log_2(1 + \text{snr}_i). \tag{1}$$

This relation, which holds for memoryless channels, links the supremum bitrate $\mathcal{C}_i$, here expressed in bit per two dimensions, to the signal to noise ratio, snr_i, experienced by each channel or subchannel i. Any reliable and implementable system must transmit at a bitrate r_i below capacity $\mathcal{C}_i$ over each subchannel, and then the margin, or SNR-gap, γ_i is introduced to analyze such systems [2, 3]

$$\gamma_i = \frac{2^{\mathcal{C}_i} - 1}{2^{r_i} - 1}. \tag{2}$$

This SNR-gap is a convenient mechanism for analyzing systems that transmit below capacity or without Gaussian input, and

$$r_i = \log_2\left(1 + \frac{\mathrm{snr}_i}{\gamma_i}\right), \tag{3}$$

with r_i the bitrate in bits per two-dimensional symbol (bits per second per subchannel) which is also the number of bits per constellation symbol.

Resource allocation is performed using loading algorithms, and diverse criteria can be invoked to decide which portion of the available resource is allocated to each of the subchannels. From an information theory point of view, the criterion is the mutual information, and the optimal resource allocation under average power constraint was first devised in [4] for Gaussian inputs and later for non-Gaussian inputs [5]. Since the performance measure is the capacity, the SNR-gap in (3) is $\gamma_i = 1$ for all i. In other cases, γ_i is higher than 1, and (3) has been exploited into many optimal and suboptimal resource allocation policies. In fact, resource allocation is a constraint optimization problem, and generally two cases are of practical interest: rate maximization (RM) and margin maximization (MM), where the objective is the maximization of the data or bitrate, and the maximization of the system margin (or power minimization in practice), respectively [6]. The MM problem gathers all non RM problems including power minimization, margin maximization (in its strict sense), and other measures such as error probabilities or goodput. (In this paper MM abbreviation is related to the general family of non-RM problems and not only to the margin maximization problem in its strict sense. The expanded form is reserved for the margin maximization in its strict sense.) It is not necessary to study all the resource allocation strategies, and equivalence or duality can be found. Families of approaches are defined, and unified processes have been used [7–10]. The loading algorithms are also split in to two families. The first is based on greedy-type approach to iteratively distribute the discrete resources [11], and the second uses Lagrangian relaxation to solve continuous resource adaptation [12]. Both approaches have been compared in terms of performance and complexity [7, 12–14]. All these adaptive resource allocations are possible when channel state information (CSI) is known at both transmitter and receiver sides. This CSI can be perfect or imperfect, and full or partial. The effects of channel estimation error and feedback delay on the performance of adaptive modulated systems can also be considered in the resource allocation process [15–17].

In this paper we shall focus henceforth on MM problems, and the main contributions are as follows:

(i) the new resource allocation algorithm and

(ii) the comparison of the different resource allocation strategies.

It is assumed that the channel estimation is perfect, and feedback CSI delay and overhead are negligible. The considered peak-power constraint, instead of the conventional average power constraint or sum power constraint, results from power mask limitation and has been taken into account in resource allocation problem [13, 18–20]. With this peak power constraint, each channel must satisfy a power constraint. Note that the sum power constraint is historically the first considered constraint [4]. Bitrate constraint comes from communication applications or service requirements, where different flows can exist, but one of them is chosen at the beginning of the communication. In this configuration, the remaining parameter to optimize is then the SNR-gap γ_i which is also related to the error probability of the communication system.

Two similar problems of MM have the same objective, that is, to maximize the system robustness. What we call robustness in this paper is the capability of a system to maintain acceptable performance with unforeseen disturbances.

The first measure of robustness is the SNR gap, or system margin, and its maximization ensures protection against unforeseen channel impairments or noise. The system margin maximization is the maximization of the minimal SNR-gap γ_i in (3) over the n subchannels. In that case the conventional equivalence between margin maximization and power minimization in MM problems is not generally true. In this paper we show that this equivalence can nevertheless be obtained in particular configurations.

The second robustness measure is the bit error rate (BER) and its minimization can reduce the packet error rate and the data retransmissions. In transmitter design, the BER minimization can be realized using uniform bit-loading and adaptive precoding [21, 22]. Analytical studies have been performed with peak-BER or average BER (computed as arithmetic mean) approaches [15, 17]. With nonuniform bit loading, the average BER must be computed as weighted arithmetic mean, and the resource allocation has been performed using a greedy-type algorithm [23]. The first main contribution of this paper is the analytical solution of the resource allocation problem in the case of weighted arithmetic mean BER minimization.

To perform the analytical study, based on a generalized Lagrangian relaxation tool, we develop a new method for finding roots of functions. This method generalizes the secant method to better fit the function-depending weight and to speed up the search of the roots. Both robustness polices are compared using a new measure. This measure evaluates the difference of the bit distributions instead of the bitrates. We also prove that both robustness policies provide the same bit distribution in asymptotic regime, which is defined for high SNR and high bitrate regimes, and this is the second main contribution in this paper. The proof is given in the case of unconstrained modulations (i.e., continuous bitrates and analytical solution) and also for QAM constellations and greedy-type algorithms. The convergence is exemplified by simulation in multicarrier communications systems.

The organization of the paper is as follows. In Section 2, the quantities to be used throughout the paper are introduced, and the robustness optimization problem is formulated in a general way for both system margin maximization and BER minimization. The equivalences between margin maximization and power minimization are worked out.

Section 3 presents the considered expressions of accurate BER, the new measure of the bit distribution differences, and the new search method of roots of functions. The solutions of formulated problems are given in Section 4 in the form of an optimum resource allocation policy based on greedy-type algorithms. The conditions of equivalence of both margin maximization and BER minimization are given in this section. Section 5 presents the analytical solution and both greedy-type and analytical methods are compared in Section 6. This Section 6 exemplifies the application of robustness optimization to multicarrier communication systems. Finally, the paper concludes in Section 7 with the proofs of several results relegated to the appendices.

Notation. The bitrates $\{r_i\}_{i=1}^n$ are defined as a number of bits per two dimensions and they are simply given by a number of bits (undertone per constellation).

2. Problem Formulation

Consider n parallel subchannels. On the i-th subchannel, the input-output relationship is

$$Y_i = h_i S_i + W_i, \tag{4}$$

where S_i is the transmitted symbol, Y_i is the received one, and h_i the complex scalar channel gain. The complex Gaussian noise W_i is a proper complex random variable with zero-mean and variance equal to $\sigma_{W_i}^2$.

The conventional average power constraint is

$$\frac{1}{n}\sum_{i=1}^{n} E\left[|S_i|^2\right] \le P, \tag{5}$$

whereas the peak-power constraint, or power spectrum density constraint, considered in this paper is

$$E\left[|S_i|^2\right] \le P, \quad i = 1, \ldots, n. \tag{6}$$

It is convenient to use normalized unit-power symbol $\{X_i\}_{i=1}^n$ such that

$$S_i = \sqrt{p_i P} X_i, \tag{7}$$

which leads to the peak-power constraint

$$p_i \le 1, \quad i = 1, \ldots, n. \tag{8}$$

It is also convenient to introduce two other variables. The first one is the conventional SNR

$$\mathrm{snr}_i = |h_i|^2 p_i \frac{P}{\sigma_{W_i}^2} \tag{9}$$

and the second is called power spectrum density noise ratio (PSDNR)

$$\mathrm{psdnr} = \frac{1}{n}\sum_{i=1}^{n} |h_i|^2 \frac{P}{\sigma_{W_i}^2}, \tag{10}$$

which is the mean signal to noise ratio over the n subchannels if and only if $p_i = 1$ for all i. This PSDNR is the ratio between the power mask at the receiver side (the transmitted power mask through the channel) and the power spectrum density of the noise. The system performance will be given according to this parameter to point out the ability of a system to exploit the available power under peak-power constraint.

Using the previous notations, (3) becomes

$$r_i = \log_2\left(1 + \frac{|h_i|^2 p_i P}{\gamma_i \sigma_{W_i}^2}\right). \tag{11}$$

With $p_i/\gamma_i = 1$ for all i, r_i is the subchannel capacity under power constraint P. With unconstrained modulations, r_i is defined in $\mathbb{R}$, but constrained modulations are used in practice and r_i takes a finite number of nonnegative values. Noninteger number of bits per symbol can also be used with fractional bit constellations [24, 25]. In this paper, modulations defined by discrete points are used with integer number of bits per symbol. Typically, $r_i \in \{0, \beta, 2\beta, \ldots, r_{\max}\}$, where β is the granularity in bits and $r_{\max}$ is the number of bits in the richest available constellation. The peak-power and bitrate constraints are then

$$p_i \le 1 \quad \forall i, \quad \sum_{i=1}^{n} r_i = R, \quad r_i \in \{0, \beta, 2\beta, \ldots, r_{\max}\} \ \forall i. \tag{12}$$

Obviously, the exploitation of the available power leads to $p_i = 1$ for all i and the constraint is simplified as

$$\sum_{i=1}^{n} r_i = R, \quad r_i \in \{0, \beta, 2\beta, \ldots, r_{\max}\} \ \forall i. \tag{13}$$

With peak-power and bitrate constraints, the resource allocation strategy is then to use all available power and to optimize the robustness.

The problem we pose is to determine the optimal bitrate allocation $\{r_i^*\}_{i=1}^n$ that maximizes a robustness measure, or inversely minimizes a frailness measure, under constraints given in (13). In its general form, this problem can be written as

$$[r_1^*, \ldots, r_n^*] = \arg \min_{\substack{\sum_{i=1}^n r_i = R \\ r_i \in \{0, \beta, 2\beta, \ldots, r_{\max}\}}} \phi\left(\{r_i\}_{i=1}^n\right), \tag{14}$$

where $\phi(\cdot)$ is the frailness measure. In this paper, this measure is given by the SNR gap or the BER. In addition to the bitrate allocation, the receiver is presumed to have knowledge of the magnitude and phase of the channel gain $\{h_i\}_{i=1}^n$, whereas the transmitter needs only to know the magnitude $\{|h_i|\}_{i=1}^n$. The objective is to find the data vector $[r_1^*, \ldots, r_n^*]$ which is the final relevant information for the transmitter. The resource allocation can then be computed on the receiver side to reduce the feedback data rate from n real numbers to n finite integer numbers. Furthermore, the integer nature of the data rates allows a full CSI at the transmitter, which is not possible with real numbers.

2.1. System Margin Maximization. The SNR-gap γ_i of the subchannel i is (3)

$$\gamma_i = \frac{\mathrm{snr}_i}{2^{r_i} - 1}. \tag{15}$$

With reliable communications, γ_i is higher than 1 for all subchannels. Let the system margin, or system SNR-gap, be the minimal value of the SNR gap in each subchannel

$$\gamma = \min_i \gamma_i. \tag{16}$$

Let γ_{init} be the initial system margin of one communication system ensuring a given QoS. Let γ be the optimized system margin of this system. Then, the system margin improvement ensures system protection in unforeseen channel impairment or noise, for example, impulse noise; bitrate and system performance targets are always reached for an unforeseen SNR reduction of $\gamma/\gamma_{\text{init}}$ over all subchannels. This robustness optimization does not depend on constellation and channel-coding types. The system margin γ is defined and optimized without knowledge of used constellations and coding, and the proposed robustness optimization works for any coding and modulation scheme.

The objective is the maximization of the system margin which is equivalent to the minimization of γ^{-1}. We note $\gamma_i(r_i)$ the function that associates r_i to γ_i. The function $\phi(\cdot)$ in (14) is then given by

$$\phi\left(\{r_i\}_{i=1}^n\right) = \max_i \frac{1}{\gamma_i(r_i)}, \tag{17}$$

$$[r_1^*, \ldots, r_n^*] = \arg \min_{\substack{\sum_{i=1}^n r_i = R \\ r_i \in \{0,\beta,2\beta,\cdots,r_{\max}\}}} \max_i \gamma_i^{-1}. \tag{18}$$

This problem is the inverse problem of bitrate maximization under peak-power and SNR-gap constraints. The solution of the bitrate maximization problem is obvious under the said constraints and given by

$$r_i^* = \beta \left\lfloor \frac{1}{\beta} \log_2\left(1 + \frac{\text{snr}_i}{\gamma_i}\right) \right\rfloor \quad \forall i. \tag{19}$$

Following the conventional SNR-gap approximation [2], the symbol error rate (SER) of QAM depending on the SNR-gap is constellation size independent with

$$\text{ser}_i(r_i) = 2\,\text{erfc}\left(\sqrt{\frac{3}{2}\gamma_i}\right) \quad \forall r_i, \tag{20}$$

where the complementary error function is usually defined as

$$\text{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt. \tag{21}$$

The system margin maximization is then equivalent to the peak-SER minimization in high-SNR regime. Note that, with (16), the system margin maximization can also be called a trough-SNR-gap maximization, and it is strongly related to the peak-power minimization. Whereas the bit-loading solution is the same for power minimization and margin maximization with sum-margin or sum-power constraints, instead of peak constraints, the following lemma gives sufficient conditions for equivalence in the case of peak constraints.

Lemma 1. *The bit allocation that maximizes the system margin under peak-power constraint* $\{p_i^{margin}\}_{i=1}^n$ *minimizes the peak-power under SNR-gap constraint* $\{\gamma_i^{power}\}_{i=1}^n$ *if* p_i^{margin} $\gamma_i^{power} = \alpha$ *for all* i.

Proof. It is straightforward using (11) and (18). Both problems have the same expression and therefore the same solution. □

This lemma provides a sufficient but not necessary condition for the equivalence of solutions, and it says that if the power and the SNR-gap constraints have proportional distributions for margin maximization and peak-power minimization problems, respectively, then both problems have the same optimal bitrate allocation. In the general case, we cannot conclude that both problems have the same solution.

2.2. BER Minimization. In communication systems, the error rate of the transmitted bits is a conventional robustness measure. By definition, the BER is the ratio between the number of wrong bits and the number of transmitted bits. With a multidimensional system, there exists several BER expressions [15, 23]. Let the BER be evaluated over the transmission of m multidimensional symbols. (We suppose that m is high enough to respect the ergodic condition and to make possible use of error probability.) In our case, the multidimensional symbols are the symbols sent over n subchannels. Let e_i be the number of erroneous bits received over subchannel i during the transmission. The BER is then given as

$$\text{ber} = \frac{\sum_{i=1}^n e_i}{m \sum_{i=1}^n r_i} = \frac{\sum_{i=1}^n r_i (e_i/m r_i)}{\sum_{i=1}^n r_i}. \tag{22}$$

The BER over subchannel i is e_i/mr_i, and the BER of n subchannels is then

$$\text{ber}\left(\{r_i\}_{i=1}^n\right) = \frac{\sum_{i=1}^n r_i \text{ber}_i(r_i)}{R} \tag{23}$$

with $\text{ber}_i(r_i)$ the function that associates the BER of channel i with the bitrate r_i. The BER of multiple variable bitrate r_i is then not the arithmetic mean of BER but is the weighted mean BER. Weighted mean BER and arithmetic mean BER are equal if $r_i = r_j$ for all i, j or if $\text{ber}_i = 0$ for all i. As there exists $\text{ber}_i \neq 0$, then weighted mean BER and arithmetic mean BER are equal if and only if $r_i = r_j$ for all i, j. Note that if the number m of transmitted multidimensional symbols depends on the subchannel i, (23) does not hold anymore. These obvious results on mean measures are not taken into account, and mean BER is erroneously used instead of mean weighted BER [15, 17].

The function $\phi(\cdot)$ in (14) is then given by

$$\phi\left(\{r_i\}_{i=1}^n\right) = \frac{1}{R} \sum_{i=1}^n r_i \text{ber}_i(r_i), \tag{24}$$

$$[r_1^*, \ldots, r_n^*] = \arg \min_{\substack{\sum_{i=1}^n r_i = R \\ r_i \in \{0,\beta,2\beta,\ldots,r_{\max}\}}} \text{ber}\left(\{r_i\}_{i=1}^n\right). \tag{25}$$

To simplify the notations, let $\mathrm{ber}(R)$ be the BER of the system. In high SNR regime with Gray mapping, $r_i \mathrm{ber}_i(r_i) = \mathrm{ser}_i(r_i)$, and then weighted mean BER can be approximated by arithmetic mean SER divided by the number of transmitted bits.

Contrary to system margin maximization, the BER minimization needs the knowledge of constellation and coding schemes, and it is based on accurate expressions of BER functions. In this paper, the used constellations are QAM, and the optimization is performed without a channel coding scheme. When dealing with practical coded systems, the ultimate measure is the coded BER and not the uncoded BER. However, the coded BER is strongly related to the uncoded BER. It is then generally sufficient to focus on the uncoded BER when optimizing the uncoded part of a communication system [26].

3. Interludes

Before solving the optimization problem, the BER approximation of QAM is presented. This approximation plays a chief role in BER minimization, and a good approximation is therefore needed. Since this paper deals with bitrate allocation, a measure of difference in the bitrate distribution is proposed and presented in this section. This section also presents a new research method of roots of functions. This method generalizes the secant method and converges faster than the secant one.

3.1. BER Approximation. Conventionally, the BER approximation of square QAM has been performed by either calculating the symbol error probability or by simply estimating it using lower and upper bounds [27]. This conventional approximation tends to deviate from the exact values when the SNR is low and cannot be applied for rectangular QAM. Exact and general closed-form expressions are developed in [28] for arbitrary one and two-dimensional amplitude modulation schemes.

An approximate BER expression for QAM can be obtained by neglecting the higher-order terms in the exact closed-form expression [28].

$$\mathrm{ber}_i \simeq \frac{1}{r_i}\left(2 - \frac{1}{I_i} - \frac{1}{J_i}\right) \mathrm{erfc}\left(\sqrt{\frac{3}{I_i^2 + J_i^2 - 2}\ \mathrm{snr}_i}\right) \tag{26}$$

with $I_i = 2^{\lfloor r_i/2 \rfloor}$, $J_i = 2^{\lceil r_i/2 \rceil}$, and $r_i = \log_2(I_i \cdot J_i)$. By symmetry, I_i and J_i can be inverted. The BER can also be expressed using the SNR-gap γ_i. Using (3) and (26), the BER is written as

$$\mathrm{ber}_i \simeq \frac{1}{r_i}\left(2 - \frac{1}{I_i} - \frac{1}{J_i}\right) \mathrm{erfc}\left(\sqrt{\frac{3(I_i J_i - 1)}{I_i^2 + J_i^2 - 2}\ \gamma_i}\right). \tag{27}$$

These two approximations allow the extension of the $\mathrm{ber}_i(r_i)$ function from $\mathbb{N}$ to $\mathbb{R}_+$ which is useful for analytical studies. Figure 1 gives the theoretical BER curves and the approximated ones from the binary phase shift keying (BPSK) to the 32768-QAM. For BER lower than $5 \cdot 10^{-2}$, the relative error is lower than 1% for all modulations.

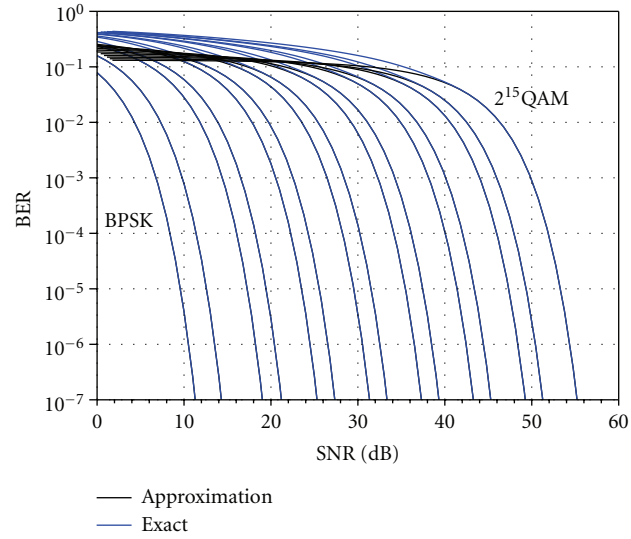


Figure 1: Exact BER curves and approximations (26).

3.2. Dissimilar Resource Allocation Measure. Two resource allocations can have the same bitrate, but this does not mean that the bitrates per subchannel are the same. To measure the difference in the bit distribution between different resource allocation strategies, we need to evaluate the *dissimilarity*. This dissimilarity measure must verify the following properties: (1) if two resource allocations lead to the same bit distribution, then the measure of dissimilarity must be null, whereas (2) if two resource allocations lead to two completely different bit distributions in loaded subchannels, then the measure of dissimilarity must be equal to one, and (3) the measure is symmetric; that is, the dissimilarity between the resource allocations X and Y must be the same as the dissimilarity between the resource allocations Y and X. We choose that the empty subchannels do not impact the measure.

Definition 2. The dissimilarity measure between the resource allocations X and Y is

$$\mu(X, Y) = \frac{\sum_{i=1}^{n} \delta(r_i(X) - r_i(Y))}{\max_{j \in \{X,Y\}} \sum_{i=1}^{n} \delta(r_i(j))}, \tag{28}$$

where $\delta(x) = 1$ if $x \neq 0$ else $\delta(x) = 0$.

This dissimilarity has the following properties.

Property 1. $\mu(X, Y) = 0$ iff $r_i(X) = r_i(Y)$ for all i.

Property 2. $\mu(X, Y) = 1$ iff $r_i(X) \neq r_i(Y)$ or $r_i(X) = r_i(Y) = 0$ for all i.

Property 3. $\mu(X, Y) = \mu(Y, X)$.

Property 4. If $\mu(X, Y) = 0$, then, for all resource allocation Z, $\mu(X, Z) = \mu(Y, Z)$.

All these properties are direct consequences of Definition 2. For a null dissimilarity, $\mu(X, Y) = 0$, all

the subchannels transmit the same number of bits, that is, $r_i(X) = r_i(Y)$ for all i. For a full dissimilarity, $\mu(X,Y) = 1$, all the nonempty subchannels of both resource allocations X and Y transmit a different number of bits, that is, for all i such as $r_i(X) \neq 0$ and $r_i(Y) \neq 0$, then $r_i(X) \neq r_i(Y)$. It is obvious that the measure is symmetric $\mu(X,Y) = \mu(Y,X)$. If two resource allocations have a null dissimilarity $\mu(X,Y) = 0$, then they are identical and for any resource allocation Z $\mu(X,Z) = \mu(Y,Z)$. The converse of this last property is not true. Note that the dissimilarity is not defined for two empty resource allocations.

For example, let $n = 4$ and $[r_1(X),\ldots,r_4(X)] = [4\ 3\ 3\ 0]$. If $[r_1(Y),\ldots,r_4(Y)] = [3\ 2\ 2\ 2]$ or $[r_1(Y),\ldots,r_4(Y)] = [5\ 5\ 0\ 0]$, then $\mu(X,Y) = 1$. If $[r_1(Y),\ldots,r_4(Y)] = [4\ 3\ 2\ 1]$, then $\mu(X,Y) = 1/2$. The measure $\mu(X,Y)$ is null if and only if $[r_1(Y),\ldots,r_4(Y)] = [4\ 3\ 3\ 0]$. The dissimilarity does not evaluate the total bitrate differences but only the bit distribution differences; the contribution of two bitrates $r_i(X)$ and $r_j(Y)$ in the dissimilarity measure is independent of the bitrate difference $|r_i(X) - r_j(Y)|$.

3.3. Generalized Secant Method. There are many numerical methods for finding roots of functions. We propose a new method, called the generalized secant method, that is, based on the secant method. This new method better fits the function-depending weight than secant method do and then improves the speed of the convergence. Before explaining this new method, a brief overview of the secant method is given.

In our case, the objective function $f(x)$ is monotonous, nondifferentiable and computable over $x \in [x_1, x_2]$ with $f(x_1)/|f(x_1)| = -f(x_2)/|f(x_2)|$. The secant method is as follows for an increasing function $f(x)$:

(1) $i = 0$, $y_0 = f(x_1)$;

(2) $x_0 = (x_2 f(x_1) - x_1 f(x_2))/(f(x_1) - f(x_2))$, $y_{i+1} = f(x_0)$;

(3) if $|y_{i+1} - y_i| \leq \epsilon$, then x_0 is the root of $f(x)$, else $\left\{ \begin{smallmatrix} y_{i+1}<0 \text{ then } x_1=x_0 \\ y_{i+1}>0 \text{ then } x_2=x_0 \end{smallmatrix} \right\}$, $i \to i+1$ and go to step 2.

The objective of the secant method is to approximate $f(x)$ by a linear function $g_i(x) = a_i x + b_i$ at each iteration i, with $g_i(x_1) = f(x_1)$ and $g_i(x_2) = f(x_2)$, and to set x_0 as the root of $g_i(x)$. The search for the root of $f(x)$ is completed when the desired precision ϵ is reached. The precision is given for y_i, but it can also be given for x_i.

As the function $f(x)$ is computable, it can be plotted and an *a posteriori* simple algebraic or elementary transcendental invertible function over $[x_1, x_2]$ can be used to better fit the function $f(x)$. The *a posteriori* information is then used to improve the search for the root. The function $f(x)$ is iteratively approximated by $a_i h(x) + b_i$ instead of $a_i x + b_i$, where $h(x)$ is the invertible function. This method is then given as follows for an increasing function $f(x)$:

(1) $i = 0$, $y_0 = f(x_1)$;

(2) $x_0 = h^{-1}((x_2 f(x_1) - x_1 f(x_2))/(f(x_1) - f(x_2)))$, $y_{i+1} = f(x_0)$;

(3) if $|y_{i+1} - y_i| \leq \epsilon$, then x_0 is the root of $f(x)$, else $\left\{ \begin{smallmatrix} y_{i+1}<0 \text{ then } x_1=x_0 \\ y_{i+1}>0 \text{ then } x_2=x_0 \end{smallmatrix} \right\}$, $i \to i+1$ and go to step 2.

Compared to the secant method, only step 2 differs and the computation of x_0 is performed taking into account the approximated shape $h(x)$ of the function $f(x)$.

This generalized secant method is used in Section 5 to find the root of the Lagrangian and is compared to the conventional secant method. In our case, $f(x)$ is the sum of logarithmic functions, and the function $h(x)$ is then the logarithmic one.

4. Optimal Greedy-Type Resource Allocations

The general problem is to find the optimal resource allocation $[r_1^*,\ldots,r_n^*]$ that minimizes $\phi(\cdot)$, the inverse robustness measure, or frailness. This is a combinatorial optimization problem or integer programming problem. The core idea in this iterative resource allocation is that a sequential approach can lead to a globally optimum discrete loading. Greedy-type methods then converge to the optimal solution. Convexity is not required for the convergence of the algorithm and monotonicity is sufficient [29]. This monotonicity ensures that the removal or addition of β bits at each iteration converges to the optimal solution. In this paper the used functions $\phi(\cdot)$ are monotonic increasing functions.

In its general form and when the objective function $\phi(\cdot)$ is not only a weighted sum function, the iterative algorithm is as follows:

(1) start with allocation $[r_1^{(0)},\ldots,r_n^{(0)}] = 0$,

(2) $k = 0$,

(3) allocate one more bit to the subchannel j for which

$$\phi\left(\left\{r_i^{(k+1)}\right\}_{i=1}^{n}\right) \tag{29}$$

is minimal, with $r_j^{(k+1)} = r_j^{(k)} + \beta$ and $r_i^{(k+1)} = r_i^{(k)}$ for all $i \neq j$,

(4) if $\sum_i r_i^{(k+1)} = R$, terminate; otherwise $k \to k+1$ and go to step 3.

The obtained resource allocation is then optimal [29] and solves (14). This algorithm needs R/β iterations. The target bitrate R is supposed to be feasible; that is, R is a multiple of β. Note that an equivalent formulation can be given starting with $r_i^{(0)} = r_{\max}$ for all i and using bit removal instead of bit addition with maximization instead of minimization. For bitrates higher than $(n/2)r_{\max}$, the number of iterations with bit removal is lower than with bit addition. The opposite is true with bitrate lower than $(n/2)r_{\max}$.

Iterative resource allocations have been firstly applied to bitrate maximization under power constraint [11]. Many works have been devoted to complexity reduction of greedy-type algorithms; see, for example, [6, 12, 30, 31] and references therein. In this section, only greedy-type algorithms are presented in order to compare the analytical resource allocation to the optimal iterative one. Note that the analytical solution can also be used as an input of the greedy-type algorithm to initialize the algorithm and to reduce the number of iterations.

4.1. System Margin Maximization. The system margin, or system SNR gap, maximization under bitrate and peak-power constraints is the inverse problem of the bitrate maximization under SNR-gap and peak-power constraints. This inverse problem has been solved, for example, in [18]. To comply with the general problem formulation, the inverse system margin minimization is presented instead of the system margin maximization.

Lemma 3. *Under bitrate and peak-power constraints, the greedy-type resource allocation that minimizes the inverse system margin* γ^{-1} (16) *allocates sequentially* β *bits to the subchannel* i *bearing* r_i *bits and for which*

$$\frac{2^{r_i+\beta}-1}{\mathrm{snr}_i} \tag{30}$$

is minimum.

Proof. It is straightforward using (17) and (29). See Appendix A for an original proof. □

The main advantage of system margin maximization is that the optimal resource allocation can be reached independently of the SNR regime. Resource allocation is always possible even for very low SNR, but it can lead to unreliable communication with SNR gap lower than 1. Lemma 3 is given with unbounded modulation orders, that is, $r_{\max} = \infty$ and $r_i \in \beta\mathbb{N}$ for all i. With full constraints (13), the subchannels that reach $r_{\max}$ are simply removed from the iterative process.

4.2. BER Minimization. The system BER minimization under bitrate and peak-power constraints is the inverse problem of bitrate maximization under peak-power and BER constraints. This inverse problem has been solved, for example, in [23]. Using (29) and (24), the solution of BER minimization is straightforward, and the corresponding greedy-type algorithm is also known as Levin-Campello algorithm [5, 32, 33]. The main drawback of this solution is that it requires good approximated BER expressions even in low-SNR regime. This constraint can be relaxed, and the following lemma gives the optimal greedy-type resource allocation for the BER minimization.

Lemma 4. *In high SNR regime and under bitrate and peak-power constraints, the greedy-type resource allocation that minimizes the BER minimizes* $(r_i+\beta)\mathrm{ber}_i(r_i+\beta)$ *at each step.*

Proof. See Appendix B. □

Lemma 4 states how to allocate bits without mean BER computation at each step. It is given without modulation order limitation. Like system margin maximization solution, the bounded modulation order is simply taken into account using $r_{\max}$ and subchannel removal.

4.3. Comparison of Resource Allocations. To compare the two optimization policies, we call $\mathcal{B}$ the resource allocation that maximizes the system margin and $\mathcal{C}$ the resource allocation that minimizes the BER. Table 1 gives an example of bitrate allocation over 20 subchannels where the SNR follows a Rayleigh distribution and with $\beta = 1$. In this example, the PSDNR defined in (10) is equal to 25 dB, and the maximum allowed bitrate per subchannel is never reached. As expected, the system margin minimization leads to a minimal SNR gap, $\min_i \gamma_i$, higher than that provided by the BER minimization policy with a gain of 0.3 dB. On the other hand, the BER minimization policy leads to BER lower than that provided by system margin minimization ($2.6 \cdot 10^{-5}$ versus $3.1 \cdot 10^{-5}$). In this example, the dissimilarity is $\mu(\mathcal{B}, \mathcal{C}) = 0.1$, and two subchannels convey different bitrates. All these results are obtained with $r_{\max} = 10$.

Table 1: Example of system margin and BER with $n = 20$, $R = 100$, psdnr = 25 dB, and $\beta = 1$.

	System margin maximization ($\mathcal{B}$)	BER minimization ($\mathcal{C}$)
$\min_i \gamma_i$	6.9 dB	6.6 dB
ber	$3.1 \cdot 10^{-5}$	$2.6 \cdot 10^{-5}$

This example shows that the difference between the resource allocation policies can be small. The question is whether both resource allocations converge and if they converge then in what cases. The following theorem answers the question.

Theorem 5. *In high-SNR regime with square QAM and under bitrate and peak-power constraints, the greedy-type resource allocation that maximizes the system margin converges to the greedy-type resource allocation that minimizes the BER.*

Proof. See Appendix D. □

The consequence of Theorem 5 is that the dissimilarity between the resource allocation that maximizes the system margin and the resource allocation that minimizes the BER is null in high-SNR regime and with square QAM. With square QAM, β should be a multiple of 2. Note that with square modulations, β can also be equal to 1 if the modulations are, for example, those defined in ADSL [34]. Figure 6 exemplifies the convergence with $\beta = 2$ as we will see later in Section 6.

5. Optimal Analytical Resource Allocations

The analytical method is based on convex optimization theory [35]. Unconstrained modulations lead to bitrates r_i defined in $\mathbb{R}$. With $r_i \in \mathbb{R}_+$ the solution is the waterfilling one. With bounded modulation order, that is, $0 \le r_i \le r_{\max}$, the solution is quite different from the waterfilling one. The solution is obtained in the framework of generalized Lagrangian relaxation using Karush-Kuhn-Tucker (KKT) conditions [36].

As the bitrates are continuous and not only integers in this analytical analysis, the constraints (13) do not hold anymore and become

$$\sum_{i=1}^{n} r_i = R\ , \quad 0 \le r_i \le r_{\max}\ \forall i. \tag{31}$$

The KKT conditions associated to the general problem (14) with (31) instead of (13) write [36]

$$-r_i \le 0\ , \quad i = 1,\dots,n, \tag{32}$$

$$r_i - r_{\max} \le 0\ , \quad i = 1,\dots,n, \tag{33}$$

$$R - \sum_{i=1}^{n} r_i = 0, \tag{34}$$

$$\mu_i \ge 0\ , \quad i = 1,\dots,n, \tag{35}$$

$$\nu_i \ge 0\ , \quad i = 1,\dots,n, \tag{36}$$

$$\mu_i r_i = 0\ , \quad i = 1,\dots,n, \tag{37}$$

$$\nu_i (r_i - r_{\max}) = 0\ , \quad i = 1,\dots,n, \tag{38}$$

$$\frac{\partial}{\partial r_i}\phi\left(\left\{r_j\right\}_{j=1}^{n}\right) - \lambda - \mu_i + \nu_i = 0\ , \quad i = 1,\dots,n, \tag{39}$$

where λ, μ_i and ν_i are the Lagrange multipliers. The first three equations (32)–(34) represent the primal constraints, (35) and (36) represent the dual constraints, (37) and (38) represent the complementary slackness, and (39) is the cancellation of the gradient of Lagrangian with respect to r_i. When the primal problem is convex and the constraints are linear, the KKT conditions are sufficient for the solution to be primal and dual optimal. For the system margin maximization problem, the function $\phi(\cdot)$ is convex over all input bitrates and SNR whereas this function is no longer convex for the BER minimization problem. Appendix C gives the convex domain of the function $\phi(\cdot)$ in the case of BER minimization problem.

The properties of the studied function $\phi(\cdot)$ are such that

$$\frac{\partial}{\partial r_i}\phi\left(\left\{r_j\right\}_{j=1}^{n}\right) = \psi_i(r_i) \tag{40}$$

is independent of r_j for all $j \ne i$. The optimal solution that solves (32)–(39) is then [36]

$$r_i^*(\lambda) = \begin{cases} 0, & \text{if } \lambda \le \psi_i(0), \\ \psi_i^{-1}(\lambda), & \text{if } \psi_i(0) < \lambda < \psi_i(r_{\max}), \\ r_{\max}, & \text{if } \lambda \ge \psi_i(r_{\max}) \end{cases} \tag{41}$$

for all $i = 1,\dots,n$ and with λ verifying the constraint

$$\sum_{i=1}^{n} r_i^*(\lambda) = R. \tag{42}$$

It is worthwhile noting that the above general solution is the waterfilling one if $r_{\max} \ge R$. The waterfilling is also the solution in the following case. Let $\mathcal{I}'$ be the subset index such that

$$\mathcal{I}' = \left\{i \mid r_i^* \ne \{0, r_{\max}\}\right\}, \tag{43}$$

and let R' the target bitrate over $\mathcal{I}'$. In this subset, $\{r_i^*\}_{i\in\mathcal{I}'}$ are solutions of

$$\begin{cases} \dfrac{\partial}{\partial r_i}\phi\left(\left\{r_j\right\}_{j=1}^{n}\right) - \lambda = 0\ , \quad \forall i \in \mathcal{I}' \\ R' - \displaystyle\sum_{i\in\mathcal{I}'} r_i(\lambda) = 0. \end{cases} \tag{44}$$

This is the solution of (14) with unbounded modulations over the subchannel index subset $\mathcal{I}'$. If $\mathcal{I}' = \{1,\cdots,n\}$ and $R' = R$, and (44) is also the solution of (14) with unconstrained modulations.

5.1. System Margin Maximization

Theorem 6. *Under bitrate and peak-power constraints, the asymptotic bit allocation which minimizes the inverse system margin is given by*

$$r_i^* = \frac{R'}{|\mathcal{I}'|} + \frac{1}{|\mathcal{I}'|}\sum_{j\in\mathcal{I}'} \log_2 \frac{\mathrm{snr}_i}{\mathrm{snr}_j}, \quad \forall i \in \mathcal{I}'. \tag{45}$$

Proof. See Appendix E. □

The solution given by Theorem 6 holds for high modulation orders which defines the asymptotic regime, compare Appendix E. If the set $\mathcal{I}'$ is known, then Theorem 6 can be used directly to allocate the subchannel bitrates. Otherwise, $\mathcal{I}'$ should be found first.

The expression of r_i^* in Theorem 6 is a function of the target bitrate R', the number $|\mathcal{I}'|$ of subchannels, and the ratios of SNR. This expression is independent of the mean received SNR or PSDNR. It does not depend on the link budget but only on the relative distribution of subchannel coefficients $\{|h_i|^2\}_{i=1}^{n}$.

5.2. BER Minimization. The arithmetic mean BER minimization has been analytically solved, for example, in [22, 37]. This arithmetic mean measure needs to employ the same number of bits per constellation which limits the system efficiency. The following theorem gives the solution of the weighted mean BER minimization that allows variable constellation sizes in the multichannel system.

Theorem 7. *Under bitrate and peak-power constraints, the asymptotic bit allocation which minimizes the BER is given by*

$$r_i^* = \frac{R'}{|\mathcal{I}'|} + \frac{1}{|\mathcal{I}'|}\sum_{j\in\mathcal{I}'} \log_2 \frac{\mathrm{snr}_i}{\mathrm{snr}_j} \quad \forall i \in \mathcal{I}' \tag{46}$$

with equal in-phase and quadrature bitrates.

Proof. See Appendix F. □

The solution given by Theorem 7 holds for high modulation orders and for subchannel BER lower than 0.1,

and these parameters define the asymptotic regime in this case, compare Appendix F. The optimal asymptotic resource allocation leads to square QAM with $\sqrt{r_i^*}$ conveyed bitrate in each in-phase and quadrature components of the signal of subchannel i. It is important to note that, in asymptotic regime, BER minimization and system margin maximization lead to the same subchannel bitrate allocation. In that case, the asymptotic regime is defined by the more stringent context which is the BER minimization. As we will see in Section 6, this asymptotic behavior can be observed when $\beta = 2$.

The main drawback of the formulas in Theorems 7 and 6 is that the subset $\mathcal{I}'$ must be known. To find this subset, the negative subchannel bitrates and those higher than $r_{\max}$ should be clipped, and $\mathcal{I}'$ can be found iteratively [18]. But clipping negative bitrates first can decrease those higher than $r_{\max}$, and clipping bitrates higher than $r_{\max}$ first can increase the negative ones. It is then not possible to apply first the waterfilling solution and after that to clip the bitrates r_i greater than $r_{\max}$ to converge to the optimal solution. Finding the set $\mathcal{I}'$ requires many comparisons, and we propose a fast iterative solution based on the generalized secant method.

5.3. Lagrangian Resolution. To solve (41), numerical iterative methods are required. It is important to observe that the function defined in (41) is not differentiable, and, thus, methods like Newton's cannot be used [18]. We use the proposed generalized secant method to better fit the function-depending weight and increase the speed of the convergence. An important point for the iterative method is that the initialization value must lead to feasible solution and should be as close as possible to the final solution.

The root of the function defined by (42) is now calculated. Let

$$f(\lambda) = \sum_{i=1}^{n} r_i(\lambda) - R. \tag{47}$$

Theorems 6 and 7 show that $r(\lambda)$ is the sum of $\log_2(\cdot)$ functions. This is the reason why the function $\log_2(\cdot)$ is used in the generalized secant method. Figure 2 shows three functions versus the parameter λ. The first function is the input function $f(\lambda)$, the second one is the function used by the generalized secant method, and the last one if the linear function used by the secant method. In this example, the common points are $\lambda = 0$ and $\lambda = 2.3$. As it is shown, the generalized secant method better fits the input function than the secant method and therefore can improve the speed of the convergence to find the root which is around $\lambda = 1/80$ in this example.

To ensure the convergence of the secant methods, the algorithm should be initialized with λ_1 and λ_2 such as $f(\lambda_1) < 0$ and $f(\lambda_2) > 0$. For both optimization problems, system margin maximization and BER minimization, the parameter λ is given by the function $\psi_i(r_i)$, and it can be

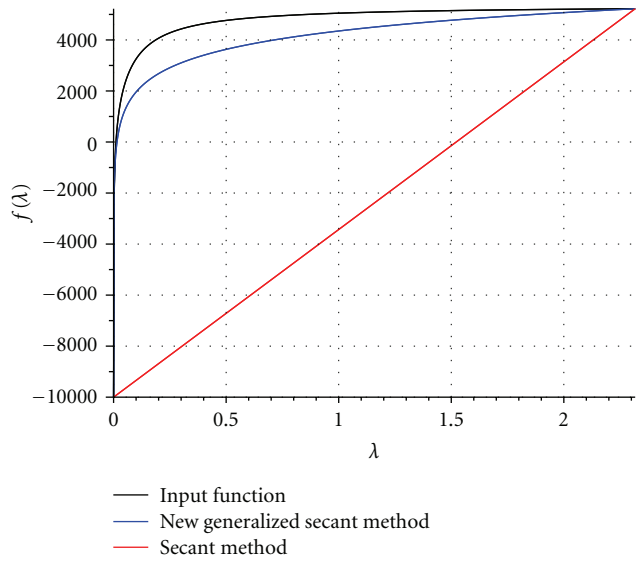


Figure 2: Approximation of the input function $f(\lambda)$ with the generalized secant method and the secant method, $n = 1024$ and $r_{\max} = 15$.

reduced to $\lambda = 2^{r_i}/\text{snr}_i$, as shown in Appendices E and F. Parameters $\{\lambda_1, \lambda_2\}$ are then chosen as

$$\lambda_1 = \frac{1}{\max_i \text{snr}_i}, \qquad \lambda_2 = \frac{2^{r_{\max}}}{\min_i \text{snr}_i}. \tag{48}$$

Using (41), $\lambda \le \lambda_1$ leads to $r_i(\lambda) = 0$ for all i, and $\lambda \ge \lambda_2$ leads to $r_i(\lambda) = r_{\max}$ for all i. Then, it follows that $f(\lambda_1) < 0$ and $f(\lambda_2) > 0$ if $R \in (0, nr_{\max})$.

Figure 3 shows the needed number of iterations for the convergence of the generalized and conventional secant methods versus the target bitrate R. Results are given over a Rayleigh distribution of the subchannel SNR with 1024 subchannels. The possible bitrates are then $R \in [0, n \times r_{\max}]$ and $\beta = 2$. Here, $r_{\max} = 15$ and then $R \le 15360$ bits per multidimensional symbol. For comparison, the number of iterations needed by the greedy-type algorithm is also plotted. Note that the greedy-type algorithm can start by empty bitrate or by full bitrate limited by $r_{\max}$ for each subchannel. The number of iterations is then given by $\min\{R, nr_{\max} - R\}$. The iterative secant and generalized secant methods are stopped when the bitrate error is lower than 1. A better precision is not necessary since exact bitrates $\{r_i\}_{i \in \mathcal{I}'}$ can be computed using Theorems 6 and 7 when $\mathcal{I}'$ is known. As it is shown in Figure 3, the generalized secant method converges faster than the secant method, except for the very low target bitrates R. For very high target bitrates, near from $n \times r_{\max}$, the number of iterations with the generalized secant method can be higher than that with the greedy-type algorithm. Except for these particular cases, the generalized secant method needs no more than 4-5 iterations to converge. In conclusion, we can say that with Rayleigh distribution of $\{\text{snr}_i\}_{i=1}^n$ and for target bitrates R such that $3\% \le R/nr_{\max} \le 97\%$, the generalized secant method converges faster than the secant method or the greedy-type algorithm.

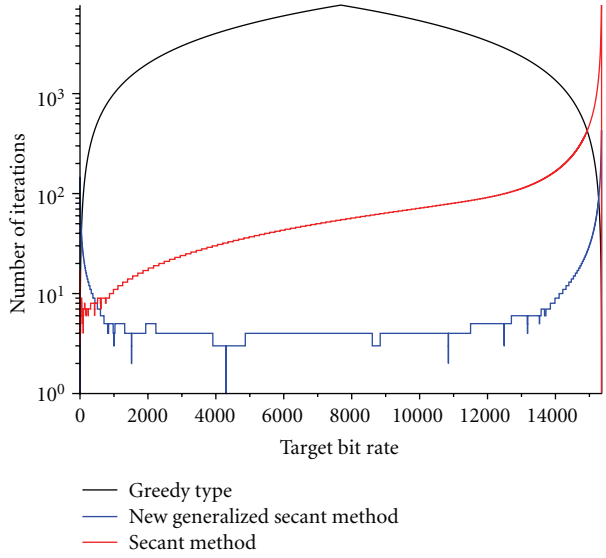


FIGURE 3: Number of iterations of the secant and generalized secant methods, and greedy-type algorithm versus the target bitrate, $n = 1024$, $r_{\max} = 15$.

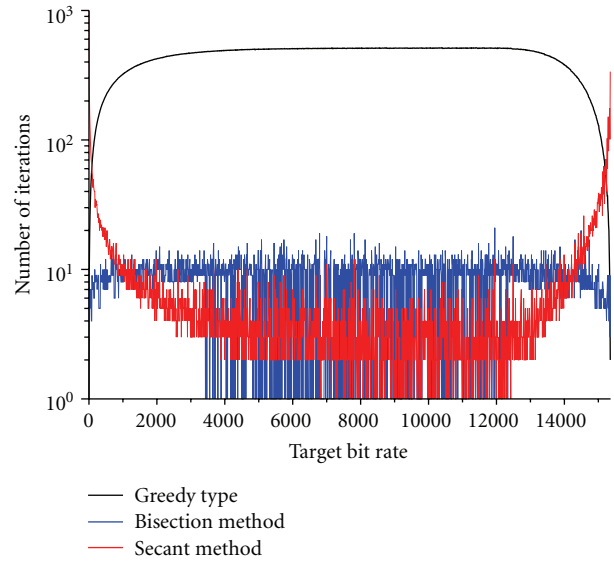


FIGURE 4: Number of iterations of the bisection and secant methods, and greedy-type algorithm for integer-bit solution versus target bitrate, $n = 1024$, $r_{\max} = 15$.

Using the generalized secant method, the bitrates are not integers and for all i, $r_i^* \in [0, r_{\max}]$. These solutions have to be completed to obtain integer bitrates.

5.4. Integer-Bit Solution. Starting from the continuous bitrate allocations previously presented, a loading procedure is developed taking into account the integer nature of the bitrates to be allotted. A simple solution is to consider the integer part of $\{r_i^*\}_{i\in\mathcal{I}'}$ and to complete by a greedy-type algorithm to achieve the target bitrate R. The integer part of $\{r_i^*\}_{i\in\mathcal{I}'}$ is then used as a starting point for the greedy algorithm. This procedure can lead to a high number of iterations. Therefore, the secant or bisection methods are suitable to reduce the number of iterations. The problem to solve is then to find the root of the following function [18]:

$$g(\alpha) = \sum_{i\in\mathcal{I}'} \lfloor r_i^* + \beta\alpha \rfloor - R', \tag{49}$$

where r_i^*, $\mathcal{I}'$, and R' are given by the continuous Lagrangian solution. This is a suboptimal integer bitrate problem, and the optimal one needs to find $\{\alpha_i\}_{i=1}^n$ instead of a unique α. As the optimal solution leads to a huge number of iterations, it is not considered. The function (49) is a nondecreasing and nondifferentiable staircase function such that $g(0) < 0$, $g(1) > 0$ because $\sum_{i\in\mathcal{I}'} r_i^* = R'$. The iterative methods can then be initialized with $\alpha_1 = 0$ and $\alpha_2 = 1$.

Two iterative methods are compared: the bisection one and the secant one. Both methods are also compared to the greedy-type algorithm. Figure 4 presents the number of iterations of the three methods to solve the integer-bit problem of the Lagrangian solution with $\beta = 1$. Results are given over a Rayleigh distribution of the subchannel SNR, with 1024 subchannels and the target bitrates are between 0 and $n \times r_{\max} = 15360$. As it is shown, the convergence is faster with bisection method than with greedy-type algorithm. For target bitrates between 10% and 90% of the maximal loadable bitrate, the secant method outperforms the bisection one with a mean number of iterations around 4 whereas the number of iterations for bisection method is higher than 8. Figure 4 also shows that $|g(0)|$ is all the time lower than the half of number of subchannels and around this value for target bitrate between 10% and 90% of the maximal loadable bitrate. Then, if the number of iterations induces by the greedy-type algorithm to solve the integer-bit problem of the Lagrangian solution that is acceptable in a practical communication system, this greedy-type completion can be used and appears to lead to the optimal resource allocation. This result obtained without proof means that the greedy-type procedure has enough bits to converge to the optimal solution. If the number of iterations induced by the greedy-type algorithm is too high (this number is around $n/2$), the secant method can be used.

The overall analytical resolution of (14) needs few iterations compared to the optimal greedy-type algorithm. Whereas the continuous solution of (14) is optimal, the analytical integer bitrate solution is suboptimal.

6. Greedy-Type versus Analytical Resource Allocations

In the previous section, the numbers of iterations of the algorithms have been compared. In this section, robustness comparison is presented and the analytical solutions obtained in asymptotic regime are also applied in nonasymptotic regime which means that $\beta = 1$ and modulation orders can be low.

The evaluated OFDM communication system is composed of 1024 subcarriers without interferences between the symbols or the subcarriers. The channel is the Rayleigh

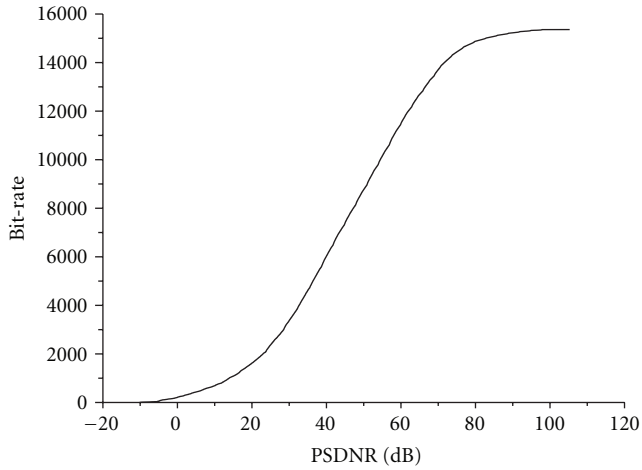


Figure 5: Target bitrate versus input PSDNR.

fading one with independent and identically distributed elements. The richest modulation order is $r_{\max} = 10$. The robustness measures are evaluated for different target bitrates which are given with the following arbitrary equation:

$$R = \left\lfloor \sum_{i=1}^{n} \min\left(\log_2\left(1 + \frac{\mathrm{snr}_i}{2}\right), r_{\max}\right) \right\rfloor. \tag{50}$$

This equation ensures reliable communications for all the input target bitrates or PSDNR. The empirical relationship between PSDNR and target bitrate is also given in Figure 5.

Figure 6 presents the output BER and the system margin of three resource allocation policies versus the target bitrate R. The first one, $\mathcal{A}$, is obtained using analytical optimization, the second, $\mathcal{B}$, is the solution of the greedy-type algorithm which maximizes the system margin, and the third, $\mathcal{C}$, is the solution of the greedy-type algorithm which minimizes the BER. Two cases are presented: one with $\beta = 1$ and the other with $\beta = 2$. All subchannel BER are lower than $2 \cdot 10^{-2}$ to use valid BER approximations. Note that, with $\beta = 1$, the system margin of allocation $\mathcal{B}$ is almost equal to 8.9 dB for all target-bit rates. This constant system margin γ is not a feature of the algorithm but is only a consequence of the relation between the target bitrate and the PSDNR.

To enhance the equivalences and the differences between the resource allocation policies, the dissimilarity is also given in Figure 6 with $\beta = 1$ and $\beta = 2$. As expected in both cases, $\beta = 1$ and $\beta = 2$, the minimal BER are obtained with allocation $\mathcal{C}$, and the maximal system margins with allocation $\mathcal{B}$.

With $\beta = 1$ and when the target bitrate increases, the Lagrangian solution converges faster to the optimal system margin maximization solution, $\mathcal{B}$, than to the optimal BER minimization solution, $\mathcal{C}$. Note that Theorem 7 is an asymptotic result valid for square QAM. With $\beta = 1$, the QAM can be rectangular, and the asymptotic result of Theorem 7 is not applicable, contrary to the result of Theorem 6 where there is not any condition on the modulation order.

The case $\beta = 2$ shows the equivalence between the optimal system margin maximization allocation and the

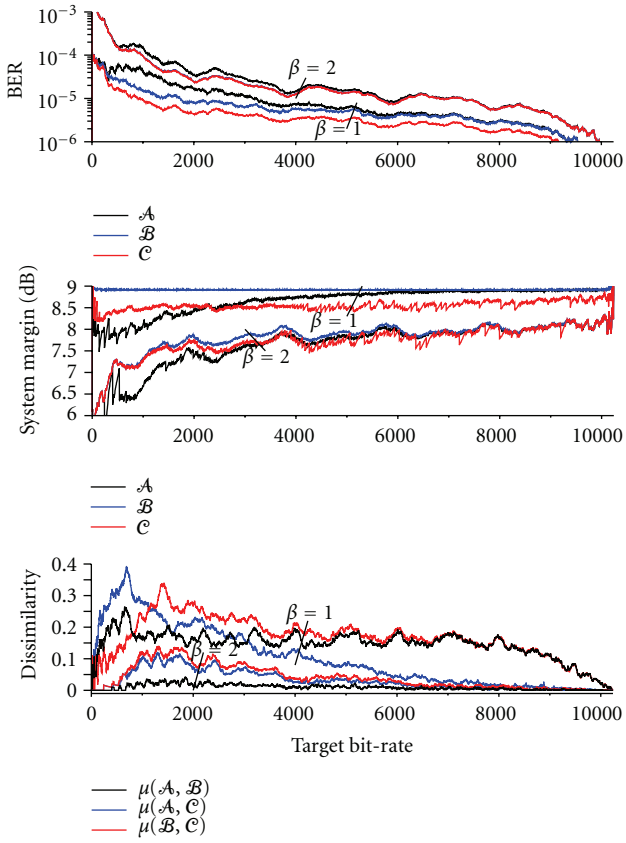


Figure 6: BER, system margin and dissimilarity versus target bitrate for Lagrangian ($\mathcal{A}$), greedy-type system margin maximization ($\mathcal{B}$) and greedy-type BER minimization ($\mathcal{C}$) algorithms, $n = 1024$, $r_{\max} = 10$, and $\beta \in \{1, 2\}$.

optimal BER minimization allocation. In this case, the asymptotic result given by Theorems 5 and 7 can be applied because the modulations are square QAM, and the convergence is ensured with high modulation orders, that is, high target bitrates. Beyond a mean bitrate per subchannel around 6, that corresponds to a target bitrate around 6000, all the allocations $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ are equivalent, and the dissimilarity is almost equal to zero. In nonasymptotic regime, the differences in BER and system margin are low. The system margin differences are lower than 1 dB, and the ratios between two BER are around 3. In practical integrated systems, these low differences will not be significant and will lead to similar solutions for both optimization policies. Therefore, these resource allocations can be interchanged.

7. Conclusion

Two robustness optimization problems have been analyzed in this paper. Weighted mean BER minimization and minimal subchannel margin maximization have been solved under peak-power and bitrate constraints. The asymptotic convergence of both robustness optimizations has been proved for analytical and algorithmic approaches. In nonasymptotic regime, the resource allocation policies

can be interchanged depending on the robustness measure and the operating point of the communication system. We have also proved that the equivalence between SNR-gap maximization and power minimization in conventional MM problem does not hold with peak-power limitation without additional conditions. Integer bit solution of analytical continuous bitrates has been obtained with a new generalized secant method, and bit-loading solutions have been compared with a new defined dissimilarity measure. The low computational effort of the suboptimal resource allocation strategy, based on the analytical approach, leads to a good tradeoff between performance and complexity.

Appendices

A. Proof of Lemma 3

We prove that the optimal allocation is reached starting from empty loading with the same intermediate loading than starting from optimal loading to empty loading. To simplify the notation and without loss of generality, $\beta = 1$.

Let $[r_1^*, \ldots, r_n^*]$ be the optimal allocation that minimizes the inverse system margin $\gamma(R^*)^{-1}$ for the target bitrate R^*, and then

$$\gamma(R^*)^{-1} = \max_i \frac{2^{r_i} - 1}{\mathrm{snr}_i}. \tag{A.1}$$

Let $[r_1, \ldots, r_n]$ be the optimal allocation that minimizes the inverse system margin $\gamma(R+1)^{-1}$ for the target bitrate $R+1 \leq R^*$. The optimal allocation for target bitrate R is obtained iteratively by removing one bit at a time from the subchannel k with the highest inverse system margin [38]

$$k = \arg \max_i \frac{2^{r_i} - 1}{\mathrm{snr}_i} \tag{A.2}$$

or

$$\frac{2^{r_k} - 1}{\mathrm{snr}_k} \geq \frac{2^{r_i} - 1}{\mathrm{snr}_i}, \quad \forall i = 1, \ldots, n. \tag{A.3}$$

The last bit removed is from the subchannel with the lowest inverse-SNR, snr_i^{-1}, because the bits over the highest inverse-SNR are first removed.

Now, let $[r_1, \ldots, r_n]$ be the optimal allocation that minimizes the inverse system margin $\gamma(R)^{-1}$ for the target bitrate $R < R^*$. Following the algorithm strategy, the optimal allocation for target bitrate $R+1$ is obtained adding one bit on subchannel j such that

$$j = \arg \min_i \frac{2^{r_i+1} - 1}{\mathrm{snr}_i}. \tag{A.4}$$

We first prove that

$$\gamma(R+1)^{-1} = \frac{2^{r_j+1} - 1}{\mathrm{snr}_j}. \tag{A.5}$$

Suppose that there exists j' such that

$$\frac{2^{r_{j'}} - 1}{\mathrm{snr}_{j'}} > \frac{2^{r_j+1} - 1}{\mathrm{snr}_j}, \tag{A.6}$$

then one bit must be added to subchannel j to obtain $r_j + 1$ bits before adding one bit to subchannel j' to obtain $r_{j'}$ bits which means that $[r_1, \ldots, r_n]$ is not optimal. As $[r_1, \ldots, r_n]$ is optimal by definition, it yields

$$\frac{2^{r_i} - 1}{\mathrm{snr}_i} \leq \frac{2^{r_j+1} - 1}{\mathrm{snr}_j} \quad \forall i = 1, \ldots, n \tag{A.7}$$

which proves (A.5). The first allocated bit is from the subchannel with the lowest inverse SNR given by (A.4) with $r_i = 0$ for all i.

Comparing (A.3) with (A.7) yields that $k = j$, and the index subchannel of the first added bit is the same as the last removed bit. All the intermediate allocations are then identical with bit-addition and bit-removal methods. There exists only one way to reach the optimal allocation R^* starting from the empty loading.

Proof of Lemma 3 can also be provided in the framework of matroid algebraic theory [19, 39].

B. Proof of Lemma 4

To simplify the notation and without loss of generality, the proof is given with $\beta = 1$. Let $[r_1, \cdots, r_n]$ be the optimal allocation for the target bitrate R such that $\sum r_i = R$. Let $R+1$ the new target bitrate. We first prove that

$$\Delta_i(r_i) = (r_i + 1)\mathrm{ber}_i(r_i + 1) - r_i \mathrm{ber}_i(r_i) \tag{B.1}$$

is a good measure at each step of the greedy-type algorithm for the BER minimization, and finally that $(r_i + 1)\mathrm{ber}_i(r_i + 1)$ can be used instead of $\Delta_i(r_i)$.

Starting from the optimal allocation of target bitrate R, the new target bitrate $R + 1$ is obtained by increasing r_j by one bit

$$\mathrm{ber}(R+1) = \frac{\left(r_j + 1\right)\mathrm{ber}_j\left(r_j + 1\right) + \sum_{i=1, i\neq j}^{n} r_i \mathrm{ber}_i(r_i)}{1 + \sum_{i=1}^{n} r_i} \tag{B.2}$$

and, using Δ_j,

$$\mathrm{ber}(R+1) = \frac{\Delta_j\left(r_j\right)}{R+1} + \frac{R}{R+1}\mathrm{ber}(R). \tag{B.3}$$

The $\mathrm{ber}(R+1)$ which is equal to $\phi(\{r_i^{(k+1)}\}_{i=1}^n)$ in (29) is minimized only if $\Delta_j(r_j)$ is minimized. The minimum $\mathrm{ber}(R+1)$ is then obtained with the increase of one bit in the subchannel j such that

$$j = \arg \min_i \Delta_i(r_i). \tag{B.4}$$

To complete the proof by induction, the relation must be true for $\mathrm{ber}(1)$. This is simply done by recalling that $\mathrm{ber}_i(0) = 0$, and then

$$\min \mathrm{ber}(1) = \min_i \mathrm{ber}_i(1) = \min_i \Delta_i(0). \tag{B.5}$$

The convergence of the algorithm to a unique solution needs the convexity of the function $r_i \mapsto r_i \mathrm{ber}(r_i)$. This convexity

is verified at high SNR. Appendix C provides a more precise domain of validity.

It remains to prove that $(r_i+1)\mathrm{ber}_i(r_i+1)$ can be used instead of $\Delta_i(r_i)$. In high SNR regime

$$\mathrm{ber}_i(r_i+1) \gg \mathrm{ber}_i(r_i) \tag{B.6}$$

and then

$$\lim_{\mathrm{snr}_i \to +\infty} \Delta_i(r_i) = (r_i+1)\mathrm{ber}_i(r_i+1) \tag{B.7}$$

which proves the lemma.

In low SNR regime, the approximation of Δ_i by $(r_i+1)\mathrm{ber}_i(r_i+1)$ remains valid; the dissimilarity between allocation using Δ_i (B.1) and allocation using $(r_i+1)\mathrm{ber}_i(r_i+1)$ is null in the domain of validity given by Appendix C.

C. Range of Convexity of $r_i\mathbf{ber}_i$

Let

$$f : \mathbb{N} \longrightarrow \mathbb{R}_+ r_i \longmapsto r_i \mathrm{ber}_i(r_i, \mathrm{snr}_i) \tag{C.1}$$

which equals the SER for high-SNR regime and Gray mapping. The function f is a strictly increasing function: $f(r_i) < f(r_i+1)$ for all snr_i, because $\mathrm{ber}(r_i, \mathrm{snr}_i) \le \mathrm{ber}(r_i+1, \mathrm{snr}_i)$ and $r_i < r_i+1$. Let $\Delta(r_i) = f(r_i+1) - f(r_i)$, and then

$$\begin{aligned}\Delta(r_i+1) - \Delta(r_i) &= f(r_i+2) - 2f(r_i+1) + f(r_i) \\ &\ge (r_i+1)(\mathrm{ber}_i(r_i+2) - 2\mathrm{ber}_i(r_i+1)).\end{aligned} \tag{C.2}$$

If $\mathrm{ber}_i(r_i+2) \ge 2\mathrm{ber}_i(r_i+1)$, then the function f is locally convex or defines a convex hull. This relation is verified for BER lower than 2×10^{-2} and for all $r_i \ge 0$.

D. Proof of Theorem 5

We prove that both metrics used in Lemmas 3 and 4 lead to the same subchannel SNR ordering. Let

$$\begin{aligned} f(r_i, \mathrm{snr}_i) &= \frac{2^{r_i+\beta}-1}{\mathrm{snr}_i}, \\ g(r_i, \mathrm{snr}_i) &= (r_i+\beta)\mathrm{ber}_i(r_i+\beta). \end{aligned} \tag{D.1}$$

We then have to prove that

$$f(r_i, \mathrm{snr}_i) \le f\left(r_j, \mathrm{snr}_j\right) \Longleftrightarrow g(r_i, \mathrm{snr}_i) \le g\left(r_j, \mathrm{snr}_j\right). \tag{D.2}$$

It is straightforward that

$$f(r_i, \mathrm{snr}_i) \le f\left(r_j, \mathrm{snr}_j\right) \Longleftrightarrow \frac{\mathrm{snr}_j}{\mathrm{snr}_i} \le \frac{2^{r_j+\beta}-1}{2^{r_i+\beta}-1}. \tag{D.3}$$

With square QAM, in high SNR regime and using (26)

$$g(r_i, \mathrm{snr}_i) = 2\left(1 - \frac{1}{\sqrt{2^{r_i+\beta}}}\right) \mathrm{erfc}\left(\sqrt{\frac{3}{2\left(2^{r_i+\beta}-1\right)}\mathrm{snr}_i}\right) \tag{D.4}$$

and it can be approximated by the following valid expression

$$g(r_i, \mathrm{snr}_i) = 2\,\mathrm{erfc}\left(\sqrt{\frac{3}{2\left(2^{r_i+\beta}-1\right)}\mathrm{snr}_i}\right). \tag{D.5}$$

Then,

$$g(r_i, \mathrm{snr}_i) \le g\left(r_j, \mathrm{snr}_j\right) \Longleftrightarrow \frac{\mathrm{snr}_j}{\mathrm{snr}_i} \le \frac{2^{r_j+\beta}-1}{2^{r_i+\beta}-1} \tag{D.6}$$

which is also given by the first inequality. In high SNR regime and with square QAM, that is, $\beta = 2$, $f(\cdot)$ and $g(\cdot)$ lead to the same subchannel SNR ordering and then

$$\arg \min_i f(r_i, \mathrm{snr}_i) = \arg \min_i g(r_i, \mathrm{snr}_i). \tag{D.7}$$

This last equation does not hold in low SNR regime (the BER approximation is not valid) or when the modulations are not square, that is, when r_i is odd. Note that (D.5) is not only a good approximation in high SNR regime, it can also be used with high modulation orders with moderate SNR regime defined in Appendix C.

E. Proof of Theorem 6

As the infinite norm is not differentiable, we use the k norm with

$$\lim_{k \to +\infty} \left(\sum_{i \in \mathcal{I}'} \gamma_i^{-k}\right)^{1/k} = \max_{i \in \mathcal{I}'} (\gamma_i^{-1}). \tag{E.1}$$

In the subset $\mathcal{I}'$, the Lagrangian of (18) for all k is

$$L_k(\{r_i\}_{i \in \mathcal{I}'}, \lambda) = \left(\sum_{i \in \mathcal{I}'} \frac{(2^{r_i}-1)^k}{\mathrm{snr}_i^k}\right)^{1/k} + \lambda\left(R' - \sum_{i \in \mathcal{I}'} r_i\right). \tag{E.2}$$

Let λ' such as

$$\lambda' = \left(\sum_{i \in \mathcal{I}'} \frac{(2^{r_i}-1)^k}{\mathrm{snr}_i^k}\right)^{(k-1)/k} \frac{\lambda}{\log 2}. \tag{E.3}$$

The optimal condition yields

$$2^{r_i}(2^{r_i}-1)^{k-1} = \mathrm{snr}_i^k \lambda'. \tag{E.4}$$

In asymptotic regime, $r_i \gg 1$ and then $2^{r_i} - 1 \simeq 2^{r_i}$. The equation of the optimal condition can be simplified and

$$r_i = \log_2(\mathrm{snr}_i) + \frac{1}{k}\log_2 \lambda'. \tag{E.5}$$

The Lagrange multiplier is to identify using the bitrate constraint, and replacing λ' in the above equation leads to the solution

$$r_i = \frac{R'}{|\mathcal{I}'|} + \frac{1}{|\mathcal{I}'|} \sum_{j \in \mathcal{I}'} \log_2 \frac{\mathrm{snr}_i}{\mathrm{snr}_j}. \tag{E.6}$$

Note that we do not need to calculate the convergence of the solution with $k \to +\infty$ to obtain the result for the infinite norm. The result holds for all values of k in asymptotic regime.

With $k = 1$, the problem is a sum SNR-gap maximization problem under peak-power constraint, and it can be solved without asymptotic regime condition. Note that this sum SNR-gap maximization problem, or sum inverse SNR-gap minimization problem, under peak-power and bitrate constraints is

$$\min \sum_{i\in\mathcal{I}'} \gamma_i^{-1} = \min_{\{r_i\}_{i\in\mathcal{I}'}} \sum_{i\in\mathcal{I}'} (2^{r_i}-1)\frac{\sigma_{W_i}^2}{|h_i|^2 P p_i} \tag{E.7}$$

and is very similar to power minimization problem under bitrate and SNR-gap constraints exchanging p_i with γ_i^{-1}

$$\min \sum_{i\in\mathcal{I}'} p_i = \min_{\{r_i\}_{i\in\mathcal{I}'}} \sum_{i\in\mathcal{I}'} (2^{r_i}-1)\frac{\sigma_{W_i}^2}{|h_i|^2 P \gamma_i^{-1}}. \tag{E.8}$$

Both problems are identical if $p_i\gamma_i = \alpha$ as it is stated by Lemma 1.

F. Proof of Theorem 7

To prove this theorem, variables I_i and J_i are used instead of r_i with

$$I_i = 2^{\lfloor r_i/2\rfloor}, \qquad J_i = 2^{\lceil r_i/2\rceil}, \tag{F.1}$$

and the bitrate constraint is

$$R = \sum_{i=1}^{n} \log_2(I_i J_i). \tag{F.2}$$

In the subset $\mathcal{I}'$, the Lagrangian of (25) is then

$$\begin{aligned} L(\{I_i,J_i\}_{i\in\mathcal{I}'},\lambda) = {} & \frac{1}{R'}\sum_{i\in\mathcal{I}'}\left(2-\frac{1}{I_i}-\frac{1}{J_i}\right) \\ & \times \operatorname{erfc}\left(\sqrt{\frac{3}{I_i^2+J_i^2-2}\operatorname{snr}_i}\right) \\ & + \lambda\left(R' - \sum_{i\in\mathcal{I}'}\log_2(I_iJ_i)\right). \end{aligned} \tag{F.3}$$

Let $X_i \in \{I_i, J_i\}$, then

$$\frac{\partial L}{\partial X_i} = X_i f(I_i,J_i) + \frac{1}{X_i^2} g(I_i,J_i) - \frac{1}{X_i}\lambda, \tag{F.4}$$

with

$$\begin{aligned} f(I_i,J_i) &= \frac{1}{R'}\left(2-\frac{1}{I_i}-\frac{1}{J_i}\right)\frac{2\sqrt{3\operatorname{snr}_i}\times e^{-3\operatorname{snr}_i/(I_i^2+J_i^2-2)}}{\sqrt{\pi}(I_i^2+J_i^2-2)^{3/2}}, \\ g(I_i,J_i) &= \frac{1}{R'}\operatorname{erfc}\left(\sqrt{\frac{3\operatorname{snr}_i}{I_i^2+J_i^2-2}}\right). \end{aligned} \tag{F.5}$$

The optimality condition yields

$$(I_i^2 - J_i^2) I_i J_i f(I_i,J_i) = (I_i - J_i) g(I_i,J_i) \ \forall i. \tag{F.6}$$

A trivial solution is $I_i = J_i$, and the other solution must verify

$$(I_i + J_i) I_i J_i f(I_i,J_i) - g(I_i,J_i) = 0. \tag{F.7}$$

To find the root of (F.7), let

$$h(x,y) = x\sqrt{y}e^{-y} - \operatorname{erfc}\left(\sqrt{y}\right) \tag{F.8}$$

with

$$\begin{aligned} x &= \frac{2}{\sqrt{\pi}}\frac{(I_i+J_i)I_iJ_i}{I_i^2+J_i^2-2}\left(2-\frac{1}{I_i}-\frac{1}{J_i}\right), \\ y &= \frac{3\operatorname{snr}_i}{I_i^2+J_i^2-2}. \end{aligned} \tag{F.9}$$

We will prove that this function is positive in a specific domain. Consider that

(1) $\sqrt{y}e^{-y} > \operatorname{erfc}(\sqrt{y})$ for $y \geq 0.334$, then for BER lower than 10^{-1}.

(2) $\sqrt{\pi}/2x > 1$ for $\{I_i, J_i\} \in [1, +\infty)^2$ and $I_i \neq 1$ or $J_i \neq 1$, and $\lim_{I_i,J_i\to 1}\sqrt{\pi}/2x = 1^+$.

Then, in the domain defined by

$$\{I_i, J_i\} \in [1, +\infty)^2 \wedge \operatorname{ber}_i \leq 0.1, \tag{F.10}$$

$h(x, y)$ is positive, and (F.7) has no solution. Thus, the only one solution of (F.6) with (F.10) is $I_i = J_i$. As we will see later the domain of (F.10) is less restrictive than the asymptotic one.

The problem is now to allocate bits with square QAM. The following upper bound is used:

$$\operatorname{ber}(r_i) = \frac{2}{r_i}\operatorname{erfc}\left(\sqrt{\frac{3\operatorname{snr}_i}{2(2^{r_i}-1)}}\right). \tag{F.11}$$

Note that this upper bound is a tight approximation with high SNR and with high modulation orders. The Lagrangian is that

$$\begin{aligned} L(\{r_i\}_{i\in\mathcal{I}'},\lambda) = {} & \frac{2}{R'}\sum_{i\in\mathcal{I}'}\operatorname{erfc}\left(\sqrt{\frac{3\operatorname{snr}_i}{2(2^{r_i}-1)}}\right) \\ & + \lambda\left(R' - \sum_{i\in\mathcal{I}'} r_i\right). \end{aligned} \tag{F.12}$$

And its derivative is

$$\frac{\partial L}{\partial r_i} = \frac{\ln 2}{\sqrt{\pi}}\frac{2^{r_i}}{2^{r_i}-1}\sqrt{\frac{3\operatorname{snr}_i}{2(2^{r_i}-1)}}e^{-3\operatorname{snr}_i/2(2^{r_i}-1)} - \lambda. \tag{F.13}$$

Let $r_i \gg 1$ for all i, then $2^{r_i} - 1 \simeq 2^{r_i}$, and the optimality condition yields

$$-\frac{3\operatorname{snr}_i}{2^{r_i}}e^{-3\operatorname{snr}_i/2^{r_i}} = -\frac{2\lambda^2\pi}{\ln^2 2}. \tag{F.14}$$

With reliable communication over the subchannel i, the Shannon's relation states that $r_i \leq \log_2(1 + \operatorname{snr}_i)$ and $3\operatorname{snr}_i/2^{r_i} \geq 3/2$ because $r_i \geq 1$. The relation between r_i and

λ is then bijective, and the real branch W_{-1} of the Lambert function [40] can be used with no possibility for confusion

$$r_i = \log_2(3\mathrm{snr}_i) - \log_2\left(-W_{-1}\left(-\frac{2\lambda^2\pi}{\ln^2 2}\right)\right). \quad \text{(F.15)}$$

With the bitrate constraint $R' = \sum_{i\in\mathcal{I}'} r_i$, we can write

$$-\log_2\left(-W_{-1}\left(-\frac{2\lambda^2\pi}{\ln^2 2}\right)\right) = \frac{R'}{|\mathcal{I}'|} - \frac{1}{|\mathcal{I}'|}\sum_{i=1}^{n}\log_2(3\mathrm{snr}_i) \quad \text{(F.16)}$$

and with (F.15)

$$r_i = \frac{R'}{|\mathcal{I}'|} + \frac{1}{|\mathcal{I}'|}\sum_{j\in\mathcal{I}'}^{n}\log_2\frac{\mathrm{snr}_i}{\mathrm{snr}_j}. \quad \text{(F.17)}$$

This result is obtained with square QAM in asymptotic regime (high modulation orders and high SNR) which is a more restrictive domain than that of (F.10).

Acknowledgments

The research leading to these results has received partial funding from the European Community's Seventh Framework Program FP7/2007-2013 under grand agreement no 213311 also referred to as OMEGA.

References

[1] A. N. Akansu, P. Duhamel, X. Lin, and M. De Courville, "Orthogonal transmultiplexers in communication: a review," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 979–995, 1998.

[2] J. M. Cioffi, "A multicarrier primer," Tech. Rep. ANSI T1E1.4/91–157, Committee Contribution, Washington, DC, USA, 1991.

[3] G. D. Forney and M. V. Eyuboglu, "Combined equalization and coding using precoding," *IEEE Communications Magazine*, vol. 29, no. 12, pp. 25–34, 1991.

[4] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the Institute of Radio Ingineers*, vol. 37, pp. 10–21, 1949.

[5] A. Lozano, A. M. Tulino, and S. Verdú, "Optimum power allocation for parallel gaussian channels with arbitrary input distributions," *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 3033–3051, 2006.

[6] T. Antonakopoulos and N. Papandreou, "Bit and power allocation in constrained multicarrier systems: the single-user case," *Eurasip Journal on Advances in Signal Processing*, vol. 2008, Article ID 643081, 14 pages, 2008.

[7] S. T. Chung and A. J. Goldsmith, "Degrees of freedom in adaptive modulation: a unified view," *IEEE Transactions on Communications*, vol. 49, no. 9, pp. 1561–1571, 2001.

[8] A. Fasano and G. di Blasio, "The duality between margin maximization and rate maximization discrete loading problems," in *Proceedings of the IEEE Workshop on Signal Processing Advances in Wireless Communications*, pp. 621–625, July 2004.

[9] D. P. Palomar and J. R. Fonollosa, "Practical algorithms for a family of waterfilling Solutions," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 686–695, 2005.

[10] I. Kim, I. S. Park, and Y. H. Lee, "Use of linear programming for dynamic subcarrier and bit allocation in multiuser OFDM," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 4, pp. 1195–1207, 2006.

[11] D. Hughes-Hartogs, *Ensemble Modem Structure for Imperfect Transmission Media*, Telebit Corporation, Cupertino, Calif, USA, 1987, US Patent 4,679,227.

[12] B. S. Krongold, K. Ramchandran, and D. L. Jones, "Computationally efficient optimal power allocation algorithms for multicarrier communication systems," *IEEE Transactions on Communications*, vol. 48, no. 1, pp. 23–27, 2000.

[13] W. J. Choi, K. W. Cheong, and J. M. Cioffi, "Adaptive modulation with limited peak power for fading channels," in *Proceedings of the 51st Vehicular Technology Conference "Shaping History Through Mobile Technologies" (VTC '00)*, pp. 2568–2572, May 2000.

[14] P. Uthansakul and M. E. Bialkowski, "Performance comparisons between greedy and Lagrange algorithms in adaptive MIMO MC-CDMA systems," in *Proceedings of the Asia-Pacific Conference on Communications*, pp. 163–167, Perth, Australia, October 2005.

[15] A. J. Goldsmith, "Variable-rate variable-power MQAM for fading channels," *IEEE Transactions on Communications*, vol. 45, no. 10, pp. 1218–1230, 1997.

[16] S. Ye, R. S. Blum, and L. J. Cimini Jr, "Adaptive OFDM systems with imperfect channel state information," *IEEE Transactions on Wireless Communications*, vol. 5, no. 11, pp. 3255–3265, 2006.

[17] N. Y. Ermolova and B. Makarevitch, "Practical approaches to adaptive resource allocation in OFDM systems," *Eurasip Journal on Wireless Communications and Networking*, vol. 2008, Article ID 160307, 2008.

[18] E. Baccarelli, A. Fasano, and M. Biagi, "Novel efficient bit-loading algorithms for peak-energy-limited ADSLtype multicarrier systems," *IEEE Transactions on Signal Processing*, vol. 50, no. 5, pp. 1237–1247, 2002.

[19] A. Fasano, "On the optimal discrete bit loading for multicarrier systems with constraints," in *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference (VTC '03)*, vol. 2, pp. 915–919, April 2003.

[20] M. A. Khojastepour and B. Aazhang, "The capacity of average and peak power constrained fading channels with channel side information," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, vol. 2, pp. 77–82, Atlanta, Ga, USA, March 2004.

[21] Y. Ding, T. N. Davidson, and K. M. Wong, "On improving the BER performance of rate-adaptive block transceivers, with applications to DMT," in *Proceedings of the IEEE Global Communications Conference*, vol. 3, pp. 1654–1658, San Francisco, Calif, USA, December 2003.

[22] D. P. Palomar, *A unified framework for communications through MIMO channels [Ph.D. thesis]*, Universitat politecnica de Catalunya, Barcelona, Spain, 2003.

[23] A. M. Wyglinski, F. Labeau, and P. Kabal, "Bit loading with BER-constraint for multicarrier systems," *IEEE Transactions on Wireless Communications*, vol. 4, no. 4, pp. 1383–1387, 2005.

[24] G. D. Forney, R. G. Gallager, G. R. Lang, F. M. Longstaff, and S. U. Qureshi, "Efficient modulation for bandlimited channels," *IEEE Journal on Selected Areas in Communications*, vol. 2, no. 5, pp. 632–647, 1984.

[25] J. M. Cioffi, *Digital Communication*, Department of Electrical Engineering, Stanford University, Stanford, Calif, USA, 2007, Course.

[26] D. P. Palomar, M. A. Lagunas, and J. M. Cioffi, "Optimum linear joint transmit-receive processing forMIMO channels with QoS constraints," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1179–1197, 2004.

[27] J. G. Proakis, *Digital Communications*, Electrical engineering, McGraw-Hill, New York, NY, USA, 3rd edition, 1995.

[28] K. Cho and D. Yoon, "On the general BER expression of one- and two-dimensional amplitude modulations," *IEEE Transactions on Communications*, vol. 50, no. 7, pp. 1074–1080, 2002.

[29] B. Fox, "Discrete optimization via marginal analysis," *Management Science*, vol. 13, no. 3, pp. 210–216, 1966.

[30] P. S. Chow, J. M. Cioffi, and J. A. C. Bingham, "Practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels," *IEEE Transactions on Communications*, vol. 43, no. 2, pp. 773–775, 1995.

[31] J. Campello, "Optimal discrete bit loading for multicarrier modulation systems," in *IEEE International Symposium on Information Theory*, p. 193, IEEE Publishing, Cambridge, Mass, USA, 1998.

[32] H. E. Levin, "A complete and optimal data allocation method for practical discrete multitone systems," in *Proceedings of the IEEE Global Communications Conference*, vol. 1, pp. 369–374, San Antonio, Tex, USA, November 2001.

[33] J. Campello, "Practical bit loading for DMT," in *Proceedings of the IEEE International Conference on Communications*, vol. 2, pp. 801–805, British Columbia, Canada, June 1999.

[34] G.992.3, *Asymmetric Digital Subscriber Line Transceivers*, ITU-T Recommendation, Geneva, Switzerland, 2002.

[35] Z.-Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, Article ID 1664998, pp. 1426–1438, 2006.

[36] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.

[37] A. Pascual-Iserte, *Channel state information and joint transmitter-receiver design in multi-antenna systems [Ph.D. thesis]*, Universitat Politecnica de Catalunya, Barcelona, Spain, 2004.

[38] L.-P. Zhu, Y. Yao, S.-D. Zhou, and S.-W. Dong, "A heuristic optimal discrete bit allocation algorithm for margin maximization in DMT systems," *Eurasip Journal on Advances in Signal Processing*, vol. 2007, Article ID 12140, 7 pages, 2007.

[39] R. J. Wilson, "An introduction to matroid theory," *The American Mathematical Monthly*, vol. 80, no. 5, pp. 500–525, 1973.

[40] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth, "On the Lambert W function," *Advances in Computational Mathematics*, vol. 5, no. 4, pp. 329–359, 1996.