



HAL
open science

An efficient and generic downlink resource allocation procedure for pre-5G networks

Alexandre Ragaleux, Sébastien Baey, Anne Fladenmuller

► **To cite this version:**

Alexandre Ragaleux, Sébastien Baey, Anne Fladenmuller. An efficient and generic downlink resource allocation procedure for pre-5G networks. *Wireless Communications and Mobile Computing*, 2016, 16 (17), pp.3089 - 3103. 10.1002/wcm.2735 . hal-01445121

HAL Id: hal-01445121

<https://hal.sorbonne-universite.fr/hal-01445121>

Submitted on 24 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RESEARCH ARTICLE

An Efficient and Generic Downlink Resource Allocation Procedure for pre-5G Networks

Alexandre Ragaleux^{1*}, Sébastien Baey¹, Anne Fladenmuller¹¹Sorbonne Universités, UPMC Univ Paris 06 / CNRS, UMR 7606, LIP6, F-75005, Paris, France.

ABSTRACT

This paper addresses the downlink resource allocation problem in pre-5G (LTE-B) networks. At each time slot, the problem is to share the radio resources between users in order to maximize a given objective function. We expressed this problem considering the LTE standard constraints, which are rarely considered in the literature. Mainly, for any given user, the base station is constrained to transmit with a single *Modulation and Coding Scheme* (MCS). We show that this problem is NP-Hard, and therefore, we propose a generic approximation algorithm which covers a large variety of objectives. This algorithm is composed of three routines that enable an effective resource sharing procedure. The first routine computes a solution for a relaxation of our main problem, while the second routine selects the most suitable MCS for each user. Finally, the last routine effectively distributes the unallocated resources. For the Max Rate policy, simulation results show that our algorithm outperforms other existing algorithms in terms of capacity, and remains close to the optimal. Under the Proportional Fairness policy, our solution also provides a very good fairness while maintaining a near-optimal capacity.

KEYWORDS

Wireless networks; LTE-B; OFDMA; opportunistic scheduling

*Correspondence

E-mail : alexandre.ragaleux@lip6.fr

1. INTRODUCTION

To keep improving the bit rates and the user experience, engineers and researchers are constantly enhancing the architecture of mobile networks. Long Term Evolution (LTE), marketed as 4G, has been widely adopted by manufacturers and the coverage of this technology is now important across the world [1]. Its successor, the 4G+ or real 4G (LTE-A, Releases 10-11) is starting to be deployed in large cities [2]. This technology offers voice compatibility and increases the theoretical peak download rate by a factor three. The last generation, the pre-5G* (LTE-B, Release 12-13) has been designed to open

*As this technology is not commercially available, the market name of LTE-B is not definitive.

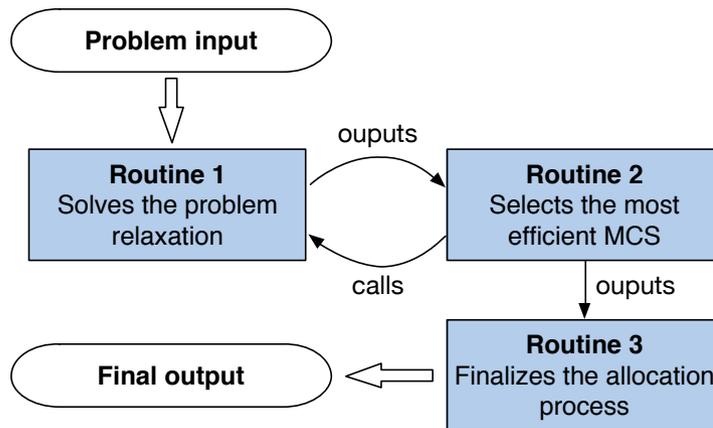


Figure 1. Operations of our proposed algorithm.

the road of the 5G by offering boosted data rates to support the new way of using cellular networks (massive data transfer). This paper is based on Release 13 of LTE-B, in which the specification on the Radio Access Network part has been released in early 2016 [3].

LTE-B networks still rely on the high performance access method OFDMA (Orthogonal Frequency Division Multiple Access) in the downlink, which allows to take full advantage of the frequency and time diversity of the radio channel. However, an efficient access method associated with suitable Modulation and Coding Schemes (MCS) are generally not sufficient to satisfy users. Indeed, the way resources are allocated has a significant impact on performance [4]. As a consequence, resource allocation algorithms for cellular networks became a fundamental research topic.

We consider a single macro-cell or a small-cell of a LTE-B network composed of one base station and a set of active mobile users. We focus on the downlink case, i.e. the communication of the base station towards the user equipments. Throughout this paper we also take the *infinite backlogs* hypothesis, i.e. we assume that the base station always has data to transmit to users. In addition, the data to be sent are organized in packets.

In LTE-B networks, the bandwidth is divided in radio resource units. At the beginning of each time slot, we tackle the resource allocation problem which consists in sharing these radio resources between users in order to maximize a given objective function. Each user experiments a different signal-to-noise ratios on each resource unit and then a fine granularity resource allocation is necessary for taking advantage of this frequency diversity. The resulting resource allocation problem is called Frequency Division Packet Scheduling (FDPS) [4].

In this paper, we take into account two uncommon constraints imposed by the LTE-B standard [3, 5]:

- *Single MCS constraint*: The base station can only transmit with a single MCS over all the resources assigned to a given user. Consequently, the instantaneous data rate per resource has to be the same.
- *Transport Block Size constraint*: For each user, the maximum amount of data that can be transmitted during a TTI is determined using a table indexed by the MCS and the number of resources assigned to it.

The single MCS constraint reduces the benefit of frequency diversity while the Transport Block Size (TBS) constraint no longer enables to express the problem linearly. As a consequence, these constraints greatly complicate the construction of efficient resource allocation algorithms (Section 2). The resulting problem is then realistic, consistent with the LTE-B standard and different from those already studied [4, 6, 7, 8] (Section 3). We call this new problem the LTE-B DL FDPS problem.

We first show that this problem is NP-Hard. Therefore, it is not possible to optimally solve an instance of the problem within the very restrictive time limit of the LTE-B system. To tackle the problem, we then propose a generic approximation algorithm which may be adapted for a wide range of scheduling objectives.

This algorithm is composed of three routines which together form a new and effective way to perform resource allocation. The first routine allows to find an approximate solution to a sub-problem which is a relaxation of our problem. The second modifies the allocation made by the previous algorithm to take into account the single MCS constraint. The last tries to effectively re-distribute unallocated resources. Figure 1 depicts the main objective of each of the routines and their relationships.

Through simulation based on the 3GPP LTE-B system, we then compare our algorithm with respect to the optimal algorithm and other algorithms from the literature, which we slightly modified in order to make them standard-compliant (see Section 5.1). The performance evaluation shows that our algorithm provides a higher system capacity than existing algorithms, remaining close to the optimal. Under the Proportional Fairness policy, a gain of 16 % is observed in terms of system capacity compared to the second best algorithm while achieving a high fairness level.

The rest of the paper is organized as follows. Section 2 describes the LTE RAN for the resource allocation problem and provides an overview of existing solutions for the DL FDPS problem in a LTE context. Section 3 defines formally the problem we consider and shows that this problem is NP-Hard. Section 4 describes our proposed algorithm. Finally, Section 5 details the simulation parameters and results. Section 6 concludes the paper.

2. LONG TERM EVOLUTION

This section defines the resource allocation problem under the LTE-B specifications and constraints. We also provide some essential symbols. As a reminder, they are listed in Table I.

2.1. LTE overview

The goal of the FDPS is to allocate the available radio resources to a set of users in order to maximize (or minimize) a given objective function. We define a *Scheduling Unit* (SU) as the smallest unit of resource that can be allocated to a user[†][3, 5]. The user set is denoted by $\mathcal{U} = \{u_1, u_2, \dots, u_U\}$, where U is the total number of users. Similarly, we define $\mathcal{N} = \{k_1, k_2, \dots, k_N\}$ the SUs set, with N the total number of SUs. Furthermore, the resource allocation process is periodic with a period called the Transmission Time Interval (TTI). At each TTI, the scheduler allocates a subset of \mathcal{N} to each user. We denote by \mathcal{N}_u the set of SUs allocated to u .

We also consider the MCS set $\mathcal{M} = \{m_1, m_2, \dots, m_M\}$, where M is the total number of MCSs. We also add a partial order \leq on \mathcal{M} such as, for $m_1, m_2 \in \mathcal{M}$ then $m_1 \leq m_2$ if and only if $m_2 = m_1$ or the spectral efficiency of m_2 is higher than the spectral efficiency of m_1 .

2.2. Problem overview

BER constraint. LTE uses Adaptive Modulation and Coding (AMC). Based on the quality of the radio channel of each user, AMC dynamically adapts the Modulation and Coding Scheme (MCS). If a user has a good quality (SNR) on a given SU then it can use a spectrally efficient MCS on it. This results in higher instantaneous rate on this resource. However, it

[†] In LTE downlink, a SU corresponds to a Resource Block Group (RBG).

increases the Bit Error Rate (BER). We consider that an MCS should not be used if the BER exceeds a given threshold. The most efficient MCS that can be used by user u on the SU k without breaking this BER constraint is denoted by $m_{u,k}$.

Problem instance: a problem instance \mathcal{I} is described by the values of the most efficient MCS over each SU and for each user:

$$\mathcal{I} = (m_{u,k})_{u \in \mathcal{U}, k \in \mathcal{N}} = \begin{pmatrix} m_{u_1, k_1} & \cdots & m_{u_1, k_N} \\ \vdots & \ddots & \vdots \\ m_{u_U, k_1} & \cdots & m_{u_U, k_N} \end{pmatrix}.$$

Single MCS (SM) constraint. Only one MCS per user can be used over all its allocated SUs. The MCS cannot be adapted to each SU in order to maximize the capacity as expected in theoretic OFDMA based systems. In the following, we assume that each user is constrained to use the MCS corresponding to its allocated SU(s) of lowest quality to meet an acceptable Bit Error Rate per SU. Consequently, we define the MCS assigned to a user $u \in \mathcal{U}$ as:

$$m_u = \min(\{m_{u,k} \mid k \in \mathcal{N}_u\}) \text{ (*minimum assumption*)}$$

Hence m_u is the MCS with the lower spectral efficiency among all its allocated SUs. This operation is not the only one possible for selecting the value of the assigned MCS (m_u). However, averaging techniques are very hard to implement in the LTE downlink [5] since the channel quality is reported to the base station as a single integer value between 0 and 31 only (*Channel Quality Indicator*). This constraint considerably complicates the resource allocation process (see Section 3).

TBS table. During a TTI, the user data is encapsulated in a Transport Block (TB). The size (in bits) of the data received by the user is called Transport Block Size (TBS). The TBS is computed according to two parameters:

- the number of SUs allocated to the user, denoted by $|\mathcal{N}_u|$ for the user u ;
- the MCS assigned to the user u , denoted by m_u .

Consequently, we denote by $TBS(n, m)$ the amount of data that can be sent from the base station to a user by allocating n SUs to it and by assigning the MCS m . $TBS(n, m)$ is determined according to a large table indexed by the two parameters. The complete TBS table can be found in [3]. Additionally, the instantaneous rate is denoted by $R(n, m) = TBS(n, m)/TTI$, where TTI refers to the length of the TTI. Most of the scheduling policies must be adapted to be consistent with this specific method of calculating the data rate. Section 5 shows an adaptation example with the Proportional Fairness (PF) policy.

Remark. Note that the Single MCS constraint does not come from the physical layer but from the standard. Indeed, the choice has been made to use a single MCS per user to avoid too much signaling overhead. Similarly, the purpose of the *Transport Block Size* table is too avoid too many rate possibilities (again, to reduce the signaling) [5, 3].

2.3. Related works and discussion

The FDPS problem is fairly new and was first introduced in [4] with the rise of OFDMA. According to this paper, the FDPS allows a very significant performance gain compared to time domain (only) schedulers. In [9], Hussam et. al have built an optimal resource allocation algorithm, but without considering the restrictive constraints imposed by the standard. Major scheduling policies for this scheme are presented in [6] and [10]. Lee et al. [11, 7] showed that the FDPS problem under MIMO configuration is NP-Hard. Moreover, they proposed an approximation algorithm for the PF policy.

Recently, Xiao et. al [12] studied the “bits-per-Joule” maximization problem with minimum rates requirements. The authors used the Lagrange dual method to reduce the combinatorial complexity and then propose a near-optimal algorithm. Similarly, Xiong et. al [13] proposed a resource allocation algorithm in multi-relay aided OFDM systems and with

| Symbol | Meaning |
|-------------------------------|---|
| U | Number of users |
| N | Number of SUs |
| M | Number of MCSs |
| \mathcal{U} | Set of users |
| \mathcal{N} | Set of SUs |
| \mathcal{N}_u | Set of SUs allocated to the user u |
| $m_{u,k}$ | Most effective MCS that can be used on SU k by the user u |
| $R(n, m)$ | Instantaneous rate with n allocated SU(s) and using the MCS m |
| m_u | Maximum MCS index that can be used by user i on \mathcal{N}_i |
| $\varphi_u(\mathcal{N}_u, m)$ | Utility function with the MCS m assigned to the user u and with \mathcal{N}_u its SU(s) set |

Table I. Main notations.

energy efficiency. This three-steps algorithm provides an efficient solution for the PF maximization problem under power constraints. However, practical LTE constraints were not considered.

In [14], the authors proposed a distributed resource allocation algorithm under a PF-based policy. They expressed the problem as a convex optimization problem which allows to build an optimal algorithm for this problem. Shajaiah et. al [15] tackled a similar problem but with carrier-aggregation, which extends the available bandwidth. The author shows that a system with two carriers aggregation is equivalent to a one-carrier system with the same amount of resources. Consequently, the same method as in [14] was used. Again, the restricting LTE particularities were not considered allowing to exploit more simply the structure of the problem.

The single MCS constraint was first tackled by Kwan et. al [16] and they proposed an heuristic by reducing the original problem to a much simpler problem. However, no performance bound is provided. A large progress has been made in [17]. The authors show that the FDPS problem with the single MCS constraint can be reduced to a sub-modular set function maximization problem. Using this result, an algorithm achieving at least half of the optimal performance was provided. This work was extended to carrier aggregation in [18], but without real algorithmic novelty. The principle of this algorithm is to iteratively allocate the resources that maximize a gain metric. This is a simple method which is common in operations research. A scheduler for the LTE-A DL MIMO was proposed in [8]. The authors expressed the problem as an Integer Linear Program (ILP) and considered its associated LP relaxation problem, i.e. without the integer constraints. The well-known simplex method was used to solve this LP relaxation problem which enables to determine an approximate solution for the original ILP problem.[‡]

However, the above previous works were mainly destined to the PF strategy. Secondly, to our knowledge, the TBS table was not considered as a part of the system. In other words, all the previous works assumed that $R(n, m) = n \times R(1, m), \forall m$. In the standard, this property is not met since the function $TBS(n, m)$ is not linear in terms of its first variable ($TBS(n_1, m) + TBS(n_2, m) \neq TBS(n_1 + n_2, m)$, with $0 \leq n_1, n_2 \leq N$ and $m \in \mathcal{M}$). The approximation made in the previous papers would be acceptable if $R(n, m) \approx n \times R(1, m)$, but this is generally not true. For example, we get from the TBS table in [3] that $TBS(1, m_1) = 16$ and $TBS(10, m_1) = 256$. Hence, $R(10, m_1)$ is about 60 %

[‡]This paper contains some flaws. The authors showed that the solution returned by the simplex algorithm is guaranteed to be integral, which is a contradiction with the fact that the problem is NP-Hard. By definition, it is not possible to find a polynomial time algorithm which solves an NP-Hard problem. Therefore, the proposed algorithm is not optimal as stated by the paper.

higher than $10 \times R(1, m_1)$ which is not negligible. In conclusion, computing the rate using the TBS table invalidates the theoretic bounds of the previous resource allocation algorithms.

3. PROBLEM FORMULATION AND HARDNESS RESULTS

In this section, we first formally define the LTE-B DL FDPS problem. Then, we introduce a sub-problem, which is a relaxation of our main problem. The introduction of this sub-problem will help us to both (1) show that the LTE-B DL FDPS problem is NP-Hard and (2) solve this same problem using state-of-the-art results (Section 4.1). To keep it simple, we present our work without considering carrier aggregation and the MIMO scheme. However, our proposed algorithm can easily be adapted to these features in the same way as in [17] and [18].

3.1. The LTE-B FDPS problem

Utility functions. In resource allocation theory, the concept of utility function is used to quantify the user-perceived benefit of a solution [19]. Our generic objective is to maximize the sum of these user-perceived benefits. The strength of this formulation is that the problem is defined in a very generic way and then covers a large variety of objectives.

Hence, in order to quantify the benefits, each user is associated to a utility function (denoted φ_u for the user u), whose parameters are (1) the resources allocated to u and (2) the MCS selected for this user:

$$\forall u, \varphi_u : 2^{\mathcal{N}} \times \mathcal{M} \rightarrow \mathbb{R}^+,$$

where $2^{\mathcal{N}}$ refers to the set of all subsets of \mathcal{N} .

We also consider that φ_u is an increasing function with respect to its two main arguments. That is, $\forall u$:

- $\forall m$, let $\varphi_{u,m}(\mathcal{N}_u) = \varphi_u(\mathcal{N}_u, m)$ then $\forall S \subseteq \mathcal{N}$ and $\forall e \in \mathcal{N} \setminus S$, $\implies \varphi_{u,m}(S \cup \{e\}) \geq \varphi_{u,m}(S)$. In other words, $\varphi_{u,m}$ is a monotone set function.
- $\forall \mathcal{N}_u \subseteq \mathcal{N}$, let $\varphi_{u,\mathcal{N}_u}(m) = \varphi_u(\mathcal{N}_u, m)$ then $\varphi_{u,\mathcal{N}_u}$ is increasing.
- $\forall m, \varphi_u(\emptyset, m) = 0$.

The LTE-B DL FDPS aims at maximizing the sum of the utility values by distributing the resources to users. In other words, the problem is to find the disjoint allocated resource sets $\mathcal{N}_{u_1}, \mathcal{N}_{u_2}, \dots, \mathcal{N}_{u_U}$ alongside with the associated single MCSs of each user $m_{u_1}, m_{u_2}, \dots, m_{u_U}$ in order to maximize $\sum_{u \in \mathcal{U}} \varphi_u(\mathcal{N}_u, m_u)$ (under the LTE constraints). More formally, the LTE-B DL FDPS problem (**P1**) is defined as follows:

Inputs: The $U \times N$ matrix $\mathcal{I} = (m_{u,k})_{u,k}$ and the expressions of the utility functions $\{\varphi_u\}_u$.

Objective:

$$\text{Maximize} \quad \sum_{u \in \mathcal{U}} \varphi_u(\{k \in \mathcal{N} \mid b_{u,k} = 1\}, m_u), \quad (1)$$

subject to:

$$\sum_u b_{u,k} \leq 1, \forall k \in \mathcal{N}, \quad (2)$$

$$m_u \leq b_{u,k} m_{u,k} + (1 - b_{u,k}) m_M, \forall u, \forall k, \quad (3)$$

$$b_{u,k} \in \{0, 1\}, \forall u, \forall k \text{ and } m_u \in \mathcal{M}, \forall u, \quad (4)$$

| | | | | | |
|--------|-----|-----|-----|-----|-----|
| User 1 | 3 | 1 | 1 | 4 | 3 |
| User 2 | 4 | 2 | 4 | 2 | 1 |
| | SU1 | SU2 | SU3 | SU4 | SU5 |

Figure 2. Example of optimal allocation with one unallocated SU.

where $b_{u,k}$ is a binary indicator which is equal to one if the SU k is allocated to the user u . Hence, $\sum_{k \in \mathcal{N}} b_{u,k}$ is equal to the number of SUs allocated to the user u . Furthermore, note that the allocated resource set of the user u is $\mathcal{N}_u = \{k \in \mathcal{N} \mid b_{u,k} = 1\}, \forall u$.

Constraint 2 ensures that each resource is allocated to at most one user. The constraint also ensures that not more than N SUs can be allocated. Indeed, it is sometimes preferable to not allocate a SU. Constraint 3 is the *single MCS* constraint and makes sure that only one MCS is used by the base station for each user. More precisely, this constraint expresses that the single MCS of each user must not lead to exceed the BER target over its allocated resources. In this way, the single MCS m_u must be less than $m_{u,k}$ if k is allocated to u . Indeed, if k is allocated to u then $b_{u,k} = 1$ and the constraint becomes $m_u \leq m_{u,k}$. Otherwise, $b_{u,k} = 0$ and the constraint becomes $m_{u,k} \leq m_M$ which is always true. Indeed, the single MCS m_u should not depend on a resource which is not allocated to u . Recall that m_u can be expressed in a non-linear way simply by $m_u = \min(\{m_{u,k} \in \mathcal{I} \mid b_{u,k} = 1\})$. This constraint has to be adapted if the minimum assumption is not made. Finally, Constraint 4 expresses that $b_{u,k}$ is a binary variable ($\forall u, k$) and m_u is in the MCS set ($m_u \in \mathcal{M}, \forall u$).

Note that, unlike the previous works presented in Section 2.3, the problem is not expressed linearly since the TBS constraint makes this formulation irrelevant.

Remark. We note that the optimal resource allocation for the LTE-B DL FDPS does not necessary allocate all SUs. This can be surprising since it is commonly accepted that the higher number of allocated resources is, the higher the throughput is. However, as only one MCS can be selected by a user, the allocation of an additional SU to this user may lower the MCS efficiency and consequently could reduce the obtained data rate. Figure 2 shows a simple example of an optimal resource allocation without allocating all SUs. In this figure, the allocation of a SU is represented by a green frame. The number inside each box represents the MCS index. The utility function is defined as the number of allocated resources multiplied by the index of the MCS that is used: $\varphi_u(\mathcal{N}_u, m_u) = |\mathcal{N}_u| \times \text{index}(m_u)$. Therefore, the optimal aggregate value is equal to 14 and leads to one unallocated resource (SU2). In practice, it follows that an efficient allocation does not necessarily allocate all SUs.

3.2. Sub-problem and hardness results

Recall that we refer to the LTE-B DL FDPS problem as **P1**.

The sub-problem. To be able to solve **P1**, we first aim at solving a “simpler” sub-problem (denoted by **P2**). We define this sub-problem in the same way as **P1** but without the single MCS constraint. Hence, **P2** is a problem relaxation. Consequently, the input remains the same as for our main problem. The two main differences between **P1** and **P2** are the following:

- since the base station can transmit with multiple MCSs to each user, the utility functions of **P2** do not depend on m_u anymore;
- the Constraint 3 is removed.

In practice, one can interpret this sub-problem as a system where each user can use a different transport block for each MCS.

The objective of **P2** is then to maximize $\sum_u \Phi_u(\mathcal{N}_u)$ subject to $\bigcap_u \mathcal{N}_u = \emptyset$ and $\mathcal{N}_u \subset \mathcal{N}, \forall u$; where Φ_u is the *single MCS free* utility function of the user u . Of course, Φ_u is closely related to φ_u as we show later in the proof of Theorem 2.

The social welfare maximization (SWM) problem. We describe here the *social welfare problem* which will help us to solve the problem **P2** using known results. The problem input is composed of:

- a set X of q items;
- a set P of p players;
- for each player i , a monotone *valuation function* $v_i : 2^X \rightarrow \mathbb{R}^+$.

The objective is to partition X into disjoint subsets S_1, \dots, S_p in order to maximize the social welfare $\sum_{i=1}^p v_i(S_i)$. This problem is NP-Hard [20].

Our results.

1. The sub-problem **P2** is NP-Hard since a reduction from the social welfare problem to **P2** can be done by building an instance of **P2** where only one MCS is available in the system ($M = 1$). It follows that our main problem (**P1**) is also NP-Hard.
2. Similarly, there is a linear-time reduction from **P2** to the social welfare maximization problem. As a consequence, to solve **P2**, it is possible to use effective algorithms from the literature which are designed to solve the social welfare problem.

Theorem 1

The sub-problem **P2** is NP-Hard.

Proof

The proof is a reduction of the social welfare problem to **P2**. Let P, X and $v_i, \forall i \in P$, an input of the social welfare maximization problem. We build the input for the sub-problem **P2** as follows:

- we consider only one MCS ($M = 1$), then $m_{u,k} = m_1, \forall (u, k)$; where m_1 refers to the only available MCS;
- The set of players is the set of users ($\mathcal{U} = P$). Similarly, the set of items is the set of SUs ($\mathcal{N} = X$);
- $\forall u \in \mathcal{U}, \mathcal{N}_u = S_u$ and $\forall S \subseteq \mathcal{N}, \varphi_u(S, m_1) = v_u(S)$.

Then, the objective is indeed to allocate the resources to the users in order to maximize $\sum_u \Phi_u(\mathcal{N}_u)$. More formally, let $C \in \mathbb{R}^+$, we claim that a partition of X is such that $\sum_{i \in \mathcal{P}} v_i(S_i) = C$ if and only if the corresponding resource affectation $\mathcal{N}_{u_1}, \dots, \mathcal{N}_{u_U}$ is such that $\sum_u \Phi_u(\mathcal{N}_u) = C$. Indeed, as only one MCS is available, we have $\Phi_u(\mathcal{N}_u) = \varphi_u(\mathcal{N}_u, m_1), \forall u$. Then, by construction $\sum_{i \in \mathcal{P}} v_i(S_i) = \sum_{u \in \mathcal{U}} \varphi_u(\mathcal{N}_u, m_1) = \sum_u \Phi_u(\mathcal{N}_u)$. In conclusion, the problem **P2** is at least as difficult as the social welfare maximization problem. \square

Corollary 1

The LTE-B DL FDPS problem (**P1**) is NP-Hard.

Proof

The proof is very similar to the previous proof and is based on a reduction of the social welfare problem to **P1**. First, let P, X , and $v_i, i \in P$ an input of the welfare maximization problem. We build the input for the problem **P1** with only one MCS ($M = 1$). As only one MCS is available, the single MCS constraint does not have to be considered. The resulting input of **P1** is then equivalent to an input of **P2**. In other words, when only one MCS is available, **P1** and **P2** are the same problems. Therefore, the remainder of the proof is the same as for the previous proof.

Consequently, our main problem is at least as difficult as the social welfare maximization problem, i.e. an optimal solution cannot be obtained using a polynomial time algorithm.

Theorem 2

There is a linear-time reduction from **P2** to the social welfare maximization problem.

Proof

First, let \mathcal{N} , \mathcal{U} and \mathcal{I} an input of the subproblem **P2**. We start by observing that the set of SUs allocated to the user u can be decomposed as follows $\mathcal{N}_u = N_{u,m_1} \cup N_{u,m_2} \cup \dots \cup N_{u,m_M}$, where $N_{u,m}$ represents the set of SUs that are allocated to u with the MCS m . It follows that each single MCS free utility function can be expressed as: $\Phi_u(\mathcal{N}_u) = \varphi_u(N_{u,m_1}, m_1) + \varphi_u(N_{u,m_2}, m_2) + \dots + \varphi_u(N_{u,m_M}, m_M)$. The objective is then to maximize:

$$\begin{aligned} \sum_{u \in \mathcal{U}} \sum_{m \in \mathcal{M}} \varphi_u(N_{u,m}, m) &= \sum_{(u,m)} \varphi_u(N_{u,m}, m) \\ &= \sum_{(u,m)} \varphi_{(u,m)}(N_{u,m}). \end{aligned}$$

Therefore, we consider each player as a pair $(u, m) \in \mathcal{U} \times \mathcal{M}$.

For each player $i = (u, m) \in \mathcal{U} \times \mathcal{M}$ we define a mapping $\pi_i : 2^{\mathcal{N}} \rightarrow 2^{\mathcal{N}}$,

$$\pi_i(S) = \pi_{(u,m)}(S) = \{k \in S \mid m \geq m_{u,k}\}. \quad (5)$$

Finally, let $P = \mathcal{U} \times \mathcal{M}$, $X = \mathcal{N}$, $S_i = N_i (\forall i \in P)$ and $v_i = \varphi_i \circ \pi_i, \forall i$. This reduction is linear with U , N and M . Then the objective is indeed to find a partition of $X = S_1 \cup S_2 \cup \dots \cup S_p$ such as $\sum_{i \in \mathcal{P}} v_i(S_i)$ is maximized. More formally, let $C \in \mathbb{R}^+$, we claim that a resource affectation $\mathcal{N}_{u_1}, \dots, \mathcal{N}_{u_U}$ is such that $\sum_u \Phi_{u \in \mathcal{U}}(\mathcal{N}_u) = \sum_{(u,m)} \varphi_{(u,m)}(N_{u,m}) = C$ if and only if the corresponding partition of $X (S_1, \dots, S_p)$ is such that $\sum_{i \in \mathcal{P}} v_i(S_i) = C$.

Consequently, each input of **P2** can be reduced to an input of the social welfare problem in linear time. This property is a strong result since we are now able to use state-of-the-art results on the social welfare problem to solve our subproblem [§]. \square

As **P1** is part of the most difficult problems, our first hope is to have a search space small enough to use a brute-force algorithm, which considers all the possibilities to find the best allocation.

Proposition 1

The size of the search space for the LTE-B DL FDPS problem (**P1**) is equal to:

$$\sum_{i=0}^{\min(U,N)} \binom{U}{i} \frac{N!}{(N-i)!} (i+1)^{N-i} M^i. \quad (6)$$

Proof

First, consider that only i active users ($0 \leq i \leq \min(U, N)$) receive SUs. The number of choices of the i scheduled users is $\binom{U}{i}$. We then assign exactly one SU to each of these users. The number of all possible allocations is then equal to the number of i -permutations in the set \mathcal{N} , which is $\frac{N!}{(N-i)!}$. The remaining SUs are then shared among all the scheduled users and another virtual user that gets the non-allocated SUs. Hence, there are $(i+1)^{N-i}$ possibilities. Finally, the number of

[§]As a matter of fact, even our main problem can be viewed as the social welfare maximization problem in a very generic way. However, in this case, the valuation functions cannot easily be classified as we will show in Subsection 4.1.

possible MCS assignments for all the scheduled users is M^i . It remains to consider all i values to obtain the total number of possible allocations:

$$\sum_{i=0}^{\min(U,N)} \binom{U}{i} \frac{N!}{(N-i)!} (i+1)^{N-i} M^i.$$

□

Hence, the number of possibilities is far too large to be browsed by a brute-force algorithm in one millisecond, even with low values for U and N . In the following, we propose an approximation algorithm to tackle our problem.

A word on Integer Linear Program (ILP) solving. To solve the LTE-B DL FDPS problem, one can think of using an ILP solver [21, 22] even if the resulting ILP problem is also NP-Hard. We expressed our problem as close as possible to an ILP. However the linear nature of our problem depends on the expression of the utility function. Unfortunately, useful utility functions are almost all non-linear. Indeed, even if we choose $\varphi_u(\mathcal{N}_u, m) = R(|\mathcal{N}_u|, m)$ (in order to maximize the capacity), we noticed that the expression of $R(n, m)$, as defined in the 3GPP standard (Section 2) is not linear (Subsection 2.3). Furthermore, even if the problem could be expressed linearly, the decision time of an LTE scheduler is, in the worst case, equal to one millisecond. Hence, time complexities of ILP solvers are too excessive for real instances.

4. APPROXIMATION ALGORITHM

In this section, we present our solution for the LTE-B DL FDPS problem. First of all, we recall the definition of an approximation algorithm. Then, after solving the sub-problem **P2** using state-of-the-art results on the social welfare problem, we present the second part of our solution which is a greedy based algorithm. We finally extend this algorithm by completing the resource allocation using a routine extension.

Definition of an approximation algorithm. Approximation algorithms are used to approximate an optimal solution of an NP-Hard problem. They provide performance guarantees compared to the optimal algorithm. A formal definition of an approximation algorithm is given below (from [23]).

Definition 1

Let P a maximization problem. For each input x of the problem P , there is a set of feasible solutions $F(x)$. For each solution $s \in F(x)$ we have a non-negative real number $c(s)$ which represents the cost of this solution. For a maximization problem, the objective is to find the solution s which maximizes $c(s)$. The optimal cost is defined by $OPT(x) = \max_{s \in F(x)} c(s)$.

Let A an algorithm which returns for an input x a feasible solution denoted by $A(x)$. A is an α -approximation ($0 \leq \alpha \leq 1$) if for each input x we have:

$$\alpha \cdot c(OPT(x)) \leq c(A(x)). \quad (7)$$

In other words, the cost of a solution computed by an α -approximation algorithm is never less than α times the cost of the associated optimal solution.

For our problem **P1**, an input is the MCS matrix $I = (m_{u,k})_{u \in \mathcal{U}, k \in \mathcal{N}}$ alongside with the expressions of the utility functions $(\varphi_u)_{u \in \mathcal{U}}$. A solution is a *resource allocation* defined by the allocated sets $\mathcal{N}_{u_1}, \dots, \mathcal{N}_{u_U}$ and the single MCS assigned to each user $(m_u)_{u \in \mathcal{U}}$. The cost of this solution corresponds to the sum of the utility values $\sum_{u \in \mathcal{U}} \varphi_u(\mathcal{N}_u, m_u)$.

4.1. On solving the sub-problem P2

We return to the sub-problem **P2** to review known results on the social welfare problem that are also applicable to **P2** using the reduction presented in the proof of Theorem 2. As the welfare maximization problem is NP-Hard, we aim at approximating the optimal solution. Recall that an α -approximation algorithm, with $\alpha \leq 1$, for the social welfare problem is an algorithm that outputs a partition $S_1 \cup \dots \cup S_p \subseteq X$ such that for any other partition $S'_1 \cup \dots \cup S'_p \subseteq X$ then $\alpha \sum_{i \in P} v_i(S'_i) \leq \sum_{i \in P} v_i(S_i)$.

However, there is no general result applicable to any kind of valuation function. This is why we restrict these functions to one of the classes presented below. These classes cover a very large number of objectives that can be targeted in an LTE system. For brevity, we do not rewrite the algorithm used to solve the SWM (*Social Welfare Maximization*) problem as it remains available in the referenced articles.

Let $f : 2^S \rightarrow \mathbb{R}^+$ a set function which can be interpreted as a valuation function in the social welfare problem or a $\varphi_{(u,m)}$ utility function in **P2** (see proof of Theorem 2).

Additive: for any $S_1 \subseteq S$ and $S_2 \subseteq S$, $f(S_1 \cup S_2) = f(S_1) + f(S_2)$. Under this property, for any subset $S' \subseteq S$, it is always possible to express $f(S')$ as a sum of weights, i.e. $f(S') = \sum_{k \in S'} w_f(k)$, where $w_f(k)$ is the weight associated to the element $k \in S'$. This is the case considered in all the previous works on resource allocation [8, 10, 17, 18]. The problem is much more easy in this scenario since there is an optimal and polynomial algorithm for the social welfare problem. Furthermore, the LTE SM-FDPS problem can be solved using a 1/2-algorithm [18].

Submodular: for any $S_1, S_2 \subseteq S$ such that $S_2 \subseteq S_1$ and for every $e \in S \setminus S_1$ then $f(S_1 \cup \{e\}) - f(S_1) \leq f(S_2 \cup \{e\}) - f(S_2)$. If the utility functions are submodular then the problem **P2** can be solved using a randomized algorithm with an approximation ratio equal to $1 - 1/e$ [24]. The submodular case may happen if the utility functions are concave, i.e. the more resources are allocated to a player (in $\mathcal{U} \times \mathcal{M}$) the less increasing is the utility value. In practice, one can imagine that a user satisfaction increases slower after receiving enough resources, i.e. the quantity $f(S' \cup \{e\})$ is growing slower as S' grows.

Subadditive: for any $S_1, S_2 \subseteq S$, $f(S_1 \cup S_2) \leq f(S_1) + f(S_2)$. This class is larger than the submodular class since every non-negative submodular function is subadditive. In this general case, to the best of our knowledge, the best approximation ratio that is achieved is 1/2 [25].

Superadditive: for any $S_1, S_2 \subseteq S$ then $f(S_1 \cup S_2) \geq f(S_1) + f(S_2)$. The problem **P1** with superadditive utility functions covers important objectives such as the MaxRate and PF policies [26]. The maximum approximation ratio for the superadditive welfare problem is $\sqrt{\log(q)}/q$ where q refers to the total number of items ($q = N$ with **P2**) [27, 28].

To solve **P2**, all utility functions have to belong to the same class. This not a real limitation since it is not possible to interpret the sum of two completely different functions. Furthermore, it is required to perform the first reduction presented in the proof of Theorem 2. Consequently, it is necessary to ensure that if φ_i ($i \in \mathcal{U} \times \mathcal{M}$) belongs to one of the above classes then $\varphi_i \circ \pi_i$ also belongs to this class, where π_i is defined in Eq. 5. As π_i simply removes the unnecessary SUs, it is straightforward that this property holds for the above classes.

The resulting first routine of our solution is described in Algorithm 1. We refer to this algorithm as `SubProblem_Algo`. This algorithm summarizes the methodology we used to compute a solution to **P2**. Indeed, this algorithm simply reduces the **P2** instance to a SWM instance. This instance is solved using one of the near-optimal algorithm described above (depending on the class of the utility functions). From Theorem 2, the performed allocation is also valid for the **P2** instance. As a consequence, the approximation factor of `SubProblem_Algo` is the factor of the algorithm used to solve the SWM instance. Furthermore, as the reduction is done in linear time, the time complexity of `SubProblem_Algo` is the same as the complexity of the state-of-the-art SWM solver we used.

- 1 **Reduce** the **P2** problem instance as a SWM problem instance (see proof of Theorem 2).
- 2 **Determine** the class of the utility functions. Then, **run** the corresponding state-of-the-art algorithm over the SWM instance.
- 3 The output, $\{N_{u,m}\}$, is also valid for the original instance of **P2**.
- 4 **return** $\{N_{u,m}\}_{u \in \mathcal{U}, m \in \mathcal{M}}$

Algorithm 1: SubProblem_Algo - Resource Allocation for **P2**.

4.2. MCS selection - greedy based algorithm

Our result depends on the algorithm used to solve the subproblem **P2** (SubProblem_Algo). For each user u and for each MCS m , SubProblem_Algo outputs $N_{u,m}$ the set of SUs assigned to u with the MCS m . Given a solution determined by the previous algorithm, we now handle the single MCS constraint.

The pseudo-code of our algorithm is presented in Algorithm. 2. We refer to this algorithm as Approx_Algo. This algorithm pre-allocates the resources using SubProblem_Algo, which attempts to allocate each SU to the user with the most effective MCS (without breaking the BER constraint). Then, for each user, the single MCS constraint is handled: the “best” MCS is selected as the value that maximizes the utility function by browsing the different subsets of its pre-allocated SUs that support at least the MCS m . After each MCS selection, the SubProblem_Algo is used again over the remaining set of SUs and the remaining users in order to re-distribute the non-allocated SUs. Finally, the pre-allocated SUs that support the “best” MCS are definitely allocated to the users.

```

1  $\forall u, \mathcal{N}_u \leftarrow \emptyset$ 
2  $\mathcal{U}^r \leftarrow \mathcal{U}$ 
   // Retrieves the solution of SubProblem_Algo (Algorithm 1):
3  $\{N_{u,m}\}_{u \in \mathcal{U}, m \in \mathcal{M}} \leftarrow \text{SubProblem\_Algo}$  over the SUs set  $\mathcal{N}$  and the user set  $\mathcal{U}$ .
   //  $\bigcup_m N_{u,m}$  is the set of all SUs pre-allocated to  $u$ .
4 for each user  $u$  in arbitrary order do
   // Select the “best” MCS:
5    $m_u \leftarrow \text{argmax}_m \varphi_u \left( \bigcup_{m' \geq m} N_{u,m'}, m \right)$ 
   // Allocation of the SUs that support the MCS  $m_u$ :
6    $\mathcal{N}_u \leftarrow \bigcup_{m \geq m_u} N_{u,m}$ 
   // Remove  $u$  from the list of users to be scheduled and re-run Algorithm 1 to compute a better resource
   // pre-allocation:
7    $\mathcal{U}^r \leftarrow \mathcal{U}^r \setminus \{u\}$ 
8    $\{N_{u,m}\}_{u \in \mathcal{U}^r, m \in \mathcal{M}} \leftarrow \text{SubProblem\_Algo}$  over the remaining SUs set  $\mathcal{N} \setminus \bigcup_u \mathcal{N}_u$  and the non-scheduled
   // users  $\mathcal{U}^r$ .
9 end
10 return  $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}}$ 

```

Algorithm 2: Approx_Algo - First allocation for **P1**.

Theorem 3

Let α the approximation factor of the algorithm used to solve the subproblem **P2**. Algorithm 2 is an $\frac{\alpha}{M}$ -approximation for the LTE-B DL FDPS problem, where M designates the total number of MCSs.

Proof

We consider a problem instance of **P1**, which is also a valid input for **P2**. First, note that the cost of our algorithm is equal to $RA = \sum_u \varphi_u(\mathcal{N}_u, m_u)$.

Let RA_1^* the cost of the optimal resource allocation for the LTE-B DL FDPS problem (**P1**) and RA_2^* the optimal cost of the sub-problem (**P2**). As **P1** corresponds to a more constrained system, it is clear that $RA_1^* \leq RA_2^*$. Since $\sum_u \sum_m \varphi_u(N_{u,m}, m)$ is the cost of the α -approximation algorithm used to solve the subproblem **P2**, then:

$$\alpha RA_1^* \leq \sum_u \sum_m \varphi_u(N_{u,m}, m) \leq RA_2^*.$$

Let u a given user. By definition of m_u we have:

$$\forall m, \forall u, \varphi_u \left(\bigcup_{m' \geq m} N_{u,m'}, m \right) \leq \varphi_u \left(\bigcup_{m' \geq m_u} N_{u,m'}, m_u \right). \quad (8)$$

Furthermore, as for a given m , φ is increasing with respect to its first parameter, then:

$$\forall u, \sum_m \varphi_u(N_{u,m}, m) \leq \sum_m \varphi_u \left(\bigcup_{m' \geq m} N_{u,m'}, m \right).$$

Hence, by summation of (8) over all possible values of m , we deduce that:

$$\forall u, \sum_m \varphi_u(N_{u,m}, m) \leq M \times \varphi_u \left(\bigcup_{m' \geq m_u} N_{u,m'}, m_u \right).$$

Consequently,

$$\begin{aligned} \alpha RA_1^* &\leq M \sum_u \varphi_u \left(\bigcup_{m' \geq m_u} N_{u,m'}, m_u \right) \\ &= M \sum_u \varphi_u(\mathcal{N}_u, m_u) \\ &\leq M \times RA, \end{aligned}$$

which completes the proof. □

Termination. Algorithm 2 (Approx_Algo) terminates since the main loop is run exactly U times.

4.3. Extension

After the MCS selection process, some of the pre-allocated SUs may be lost if they do not support the “best” MCS. Therefore, the above greedy algorithm can lead to a waste of resources.

To address this issue, we propose an extension to Algo. 2. This extension is presented in Algo. 3.

In this extension, each user is scheduled at most once. Hence, the algorithm maintains the set \mathcal{U}^r of users that remain to be scheduled. The allocation made by the approximation algorithm $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{N}_u}$ is first retrieved. The set of unallocated SUs is denoted \mathcal{N}^r and the set \mathcal{U}^r is initialized with all users whose utility value can be improved, i.e.:

$$\begin{aligned} \mathcal{U}^r &\triangleq \{u \in \mathcal{U} \mid \exists X \subset \mathcal{N}^r, \exists m \in \mathcal{M}, \\ &\quad \varphi_u(X \cup \mathcal{N}_u, m) > \varphi_u(\mathcal{N}_u, m_u)\}. \end{aligned} \quad (9)$$

Let $u \in \mathcal{U}^r$ a user and m a MCS. We now denote by $N_{u,m}$ the subset of \mathcal{N}^r such that for each $k \in N_{u,m}$ we have $m \leq m_{u,k} \leq m_u$. Note that the sets $N_{u,m}$ are not necessarily disjoint between users. We define the *gain per resource* of the user u as follows:

$$g_u = \max_m \frac{\varphi_u(N_u \cup N_{u,m}, m) - \varphi_u(N_u, m_u)}{|N_{u,m}|}$$

and m'_u the value of m for which g_u is maximized.

Hence, we consider the elected user $v \in \mathcal{U}$ as the user with the highest *gain per resource* metric:

$$v = \operatorname{argmax}_u g_u.$$

Our extension finally allocates all SUs in N_{v,m'_v} and assigns the MCS m'_v to v . This election process is then repeated after updating \mathcal{U}^r according to (9). The past elected user v necessarily does not belong to the remaining users set.

```

1   $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}} \leftarrow \text{Approx\_Algo}()$ ;
2   $\mathcal{N}^r \leftarrow \mathcal{N} \setminus \bigcup_u \mathcal{N}_u$  //  $\mathcal{N}^r$  is the non-allocated resource set.
3   $\mathcal{U}^r \leftarrow$  compute the remaining users set according to Eq. (9)
4  while  $\mathcal{U}^r$  is not empty do
5      foreach user  $u \in \mathcal{U}$  do
6          // Compute the set of all SUs that can be used with MCS  $m$ .
7           $\forall m \leq m_u, N_{u,m} \leftarrow \{k \in \mathcal{N}^r \mid m_{u,k} \leq m\}$ 
8          // Compute  $g_u$ , the "gain per resource" of the user  $u$ .
9           $\forall m \leq m_u, g_u(m) \leftarrow \frac{\varphi(N_{u,m} \cup \mathcal{N}_u, m) - \varphi(\mathcal{N}_u, m_u)}{|N_{u,m}|}$ 
10          $g_u \leftarrow \max_m g_u(m)$ 
11         // Compute  $m'_u$  the MCS associated to  $g_u$ .
12          $m'_u \leftarrow \operatorname{argmax}_m g_u(m)$ 
13     end
14     // Find the user  $v$  with the highest gain per resource metric.
15      $v \leftarrow \operatorname{argmax}_u g_u$ 
16      $m_v \leftarrow m'_v$ 
17     // Allocate to user  $v$  the available resources that support the MCS  $m'_v$ .
18      $\mathcal{N}_v \leftarrow \mathcal{N}_v \cup N_{v,m_v}$ 
19      $\mathcal{U}^r \leftarrow$  apply Eq. (9)
20 end
21 return  $\{\mathcal{N}_u, m_u\}_{u \in \mathcal{U}}$ 

```

Algorithm 3: Extension.

4.4. A word on complexity

For the class of functions studied here, the temporal complexity of `SubProblem_Algo` is $\mathcal{O}(\delta(U, M, N))$, where δ is a polynomial function. In order to find the best MCS for every user, Algorithm 2 browses all MCSs for each user. Furthermore, the `SubProblem_Algo` is run after each MCS selection. This leads to a complexity of $\mathcal{O}(U(\delta(U, M, N) + M))$. Regarding the extension (Algorithm 3), the maximum number of entries in the internal loop is equal to $\sum_{i=1}^U i$. In this loop, all MCSs can be considered in order to find the largest $g_u(m)$ value. Furthermore, for each user and for each MCS, the set $N_{u,m}$ needs to be determined. Thus, the complexity of the complete resource allocation procedure is equal to $\mathcal{O}(U \cdot \delta(U, M, N) + U^2 MN)$, which is suitable for LTE.

| Parameter | Value |
|---------------------------|--------------------------------|
| Channel Model | Path loss with Rayleigh fading |
| Reference distance | $1 d_{ref}$ |
| SNR to CQI mapping method | EESM [30] |
| Duplexing mode | Time Division Duplex (TDD) |
| TDD configuration | Configuration 5 [5] |
| Number of users | 10 |
| Initial distance of users | $1 d_{ref}$ |
| Mobility model | Random Way Point |
| Mobility area | Circle of radius $2 d_{ref}$ |
| Traffic model | Infinite backlogs |

Table II. Simulation parameters.

5. PERFORMANCE EVALUATION

The objective of this section is to compare, by simulation, the performance of our approximation algorithm (with or without its extension), to other resource allocation procedures.

Even if our work can be used for a large variety of other objectives, we expressed the utility functions for two particular scheduling policies:

- The Max Rate policy (Section 5.2), which aims at maximizing the capacity;
- The Proportional Fairness (PF) policy (Section 5.3), which allows a trade-off between the capacity and the fairness between users.

Indeed, to run simulations, we need to concretely express the utility functions.

In both policies, our utility functions can be expressed as $\varphi_u(\mathcal{N}_u, m_u) = w_u R(|\mathcal{N}_u|, m_u) = w_u TBS(|\mathcal{N}_u|, m_u) / TTI$, where $w_u = 1$ for the MaxRate policy and $0 < w_u < 1$ for the PF policy. Since $\forall m f_m(S) = TBS(|S|, m)$ can be considered as superadditive[¶], it results that these two policies are also superadditive.

5.1. Simulation Framework

We used the discrete-event simulator Riverbed Modeler (formerly known as Opnet Modeler) [29] in which we have implemented our own LTE module. The main simulation parameters are given in Table II.

The distance d_{ref} is chosen so that the corresponding attenuation a_{ref} leads to an SNR equal to 31 dB:

$$SNR_{ref} = 10 \log_{10} \left(\frac{P_{max} T_s}{N_0} \times a_{ref} \right) = 31 \text{ dB.}$$

Hence, the SNR experienced by the user u (at a distance d_u from the base station) on the subcarrier j is given by:

$$SNR_{u,j} = SNR_{ref} + 10 \log_{10} \left(\left(\frac{d_{ref}}{d_u} \right)^{3.5} \times \gamma_{u,j} \right),$$

[¶]As a matter of fact, there are few counter-examples in the TBS table that invalidates the superadditive property. However, as one SU corresponds to several RBs, then the number of counter-examples is very low. The property is almost always observed and always holds when the number of allocated SUs is low (which is the most important). In the end, the consideration of the superadditive property brings much higher performances than if we were considering the additive property.

| MCS index | Modulation | Code rate | β_m | SNR threshold (dB) |
|-----------|------------|-----------|-----------|--------------------|
| 1 | QPSK | 0.3 | 1.48 | 2.90 |
| 2 | QPSK | 0.58 | 1.62 | 5.15 |
| 3 | 16QAM | 0.36 | 3.10 | 7.12 |
| 4 | 16QAM | 0.48 | 4.32 | 12.10 |
| 5 | 64QAM | 0.45 | 7.71 | 15.85 |
| 6 | 64QAM | 0.65 | 19.6 | 19.25 |

Table III. β_m and SNR threshold values for each MCS [31, 32].

where $\gamma_{j,n}$ is a Rayleigh random variable with mean equal to 1. To avoid too much overhead, each user u transmits to the base station the indexes of the most efficient MCS on each SU k without breaking the BER constraint, i.e. $\{m_{u,k}\}_{k \in \mathcal{N}}$.

In our simulation, each SU is composed of 3 RBs (Resource Blocks). One RB corresponds to 12 subcarriers during one subframe. Among these 12 subcarriers, only 4 can be used to estimate the SNR [5]. Thus, on a whole SU, each mobile estimates exactly 12 SNR values.

We used the *Exponential Effective S(IN)R Mapping* (EESM) [31] method to compute the $m_{u,k}$ values as a function of these 12 SNR values (see [31, 32, 33] for further details on EESM). The SNR thresholds above which the corresponding MCS can be used without exceeding a BER of 10^{-3} are listed in Table III. This table also provides the β_m parameter used in the EESM method [31]. Finally, as we used the TDD mode, the base station is able to build the matrix \mathcal{I} at each uplink subframe (every 5 ms with the TDD Configuration 5).

Our approximation algorithm with its extension (*approx + ext*) is compared with:

- *approx*, the same approximation algorithm but without the extension (in order to evaluate the gain of this extension);
- the *classic algorithm* [26], which does not optimize the allocations according to the single MCS constraint and the TBS table, even though we made sure that these specificities are respected in the end;
- USA (*Unified Scheduling Algorithm*) detailed in [18], which considers the SM constraint but does not compute the rates using the TBS table;
- the *optimal algorithm* which finds the optimal solution.

The classic algorithm and USA are based on per-SU metrics since the utility functions have been assumed additive. We adapted these algorithms to compute the metric $\lambda_{u,k}$ of the user u over the SU k as $\lambda_{u,k} = w_u \times R(1, m_{u,k})$, where w_u depends on the policy. After the allocation process, we computed the resulting instantaneous rate of the user u as $R(n_u, m_u)$, where n_u is the number of SUs allocated to u and m_u is the assigned MCS.

Furthermore, as our problem is not linear (as formulated), we could not use an ILP (Integer Linear Program) solver to find the optimal allocation. As a consequence, the optimal allocation takes a very long time to compute (optimal algorithm of exponential complexity). This is why, to compare our algorithm to the optimal one, we first simulate with 10 users and we limit the number of SUs to 18.

To evaluate the fairness among users we used Jain's Fairness index:

$$F(\Delta t) = \frac{(\sum_{u \in \mathcal{U}} \bar{r}_u(\Delta t))^2}{U \cdot \sum_{u \in \mathcal{U}} (\bar{r}_u(\Delta t))^2}, \quad (10)$$

where $\bar{r}_u(\Delta t)$ is the rate of the user u obtained over a time window of duration Δt . This index ranges from $1/U$ (poor fairness) to 1 (perfect fairness).

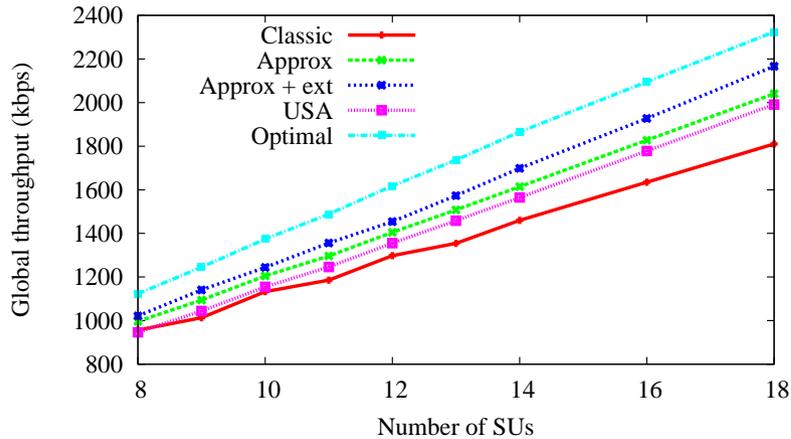


Figure 3. Capacity with MaxRate.

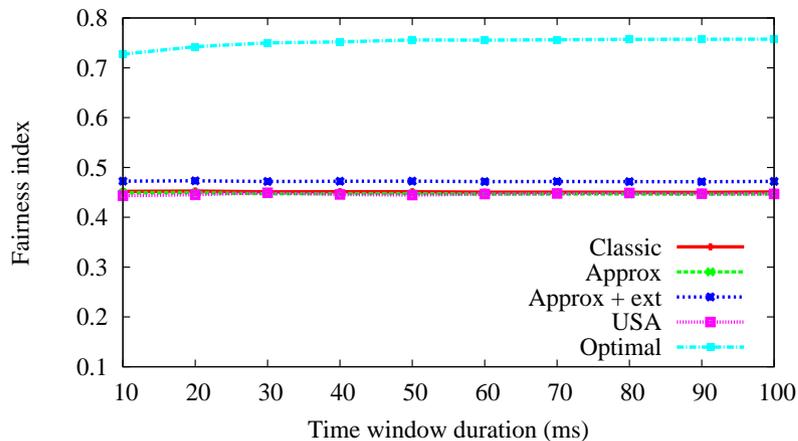


Figure 4. Fairness with MaxRate.

5.2. Scenario 1 : Max Rate policy

In this first scenario, we aim at maximizing the capacity, i.e. we express the utility function as follows: $\varphi(n_u, m_u) = R(n_u, m_u)$ for a given user u and at a given time slot and with $n_u = |\mathcal{N}_u|$. Recall that $R(n, m)$ is the rate achieved with n SUs and with the MCS m .

Fig. 3 shows the maximum average system load as a function of the number of SUs. As expected, the curves are linear with the number of SUs. Our solution achieves the best results among sub-optimal algorithms. Furthermore, our algorithm with its extension provides even better results: the slope of its associated curve appears to be similar to the optimal solution. The gap with respect to the other algorithms seems to be increasing with the number of SUs. Indeed, as the number of SUs increases, the probability of having non-allocated SUs with Algorithm 2 is higher. Then, the beneficial effect of our extension is even stronger. Fig. 4 shows the Jain's fairness index as a function of the time window duration. As fairness is not the target of this policy then (1) the fairness indexes are constant and (2) fairness indexes are low for all algorithms except for the optimal algorithm.

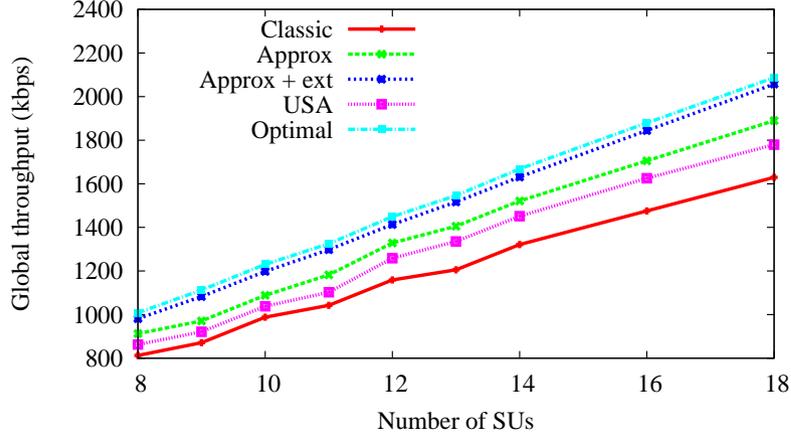


Figure 5. Capacity with PF.

5.3. Scenario 2 : Proportional Fairness policy

Here, the objective is to get a fairer resource allocation without completely compromising the data rates of the users. In this aim, we modified the traditional PF policy. In traditional systems (without FDPS and without SM constraint) the objective of the PF schedulers is to maximize $\sum_{u \in \mathcal{U}} \log R_u$, where R_u is the long time service rate of the user u . In [26], it was proved that maximizing this objective is equivalent to maximizing $\sum_u r_u(t)/R_u(t)$, where $r_u(t)$ is the rate of the user u at the t^{th} TTI and $R_u(t)$ is the *mean rate per time slot* of u .

$$R_u(t+1) = \begin{cases} (1 - \beta)R_u(t) + \beta r_u(t) & \text{if } u \text{ is scheduled} \\ & \text{on the time slot } t, \\ (1 - \beta)R_u(t) & \text{otherwise,} \end{cases}$$

where $0 \leq \beta \leq 1$.

Our last contribution is to extend the PF policy in order to take into account the single MCS constraint and the specificities of the FDPS. We start by redefining $R_u(t)$ as the *mean rate per SU*:

$$R_u(t+1) = \begin{cases} (1 - \beta)R_u(t) + \beta R(n_u(t), m_u(t))/n_u(t) & \text{if } n_u(t) \neq 0, \\ (1 - \beta)R_u(t) & \text{otherwise,} \end{cases}$$

where β is an averaging factor generally set to 10^{-4} [7, 26]. Then, we express the SM-PF (Single MCS Proportional Fairness) policy by defining the following utility function:

$$\varphi_u(\mathcal{N}_u, m_u) = \frac{R(n_u(t), m_u(t))}{n_u(t) \times R_u(t)}.$$

This utility function leads to a good trade-off between fairness and capacity by allocating resources only if the resulting throughput is better than its average.

Fig. 5 shows the average capacity as a function of the number of resources while Fig. 6 shows the fairness index as a function of the time window duration. The average throughput of each suboptimal algorithm is clearly lower than the one obtained with our algorithm. Of course, the optimal algorithm gets the best overall throughput and the best fairness index. The Jain's fairness of our complete allocation process is still very good (around 0.94). The reason why our algorithm

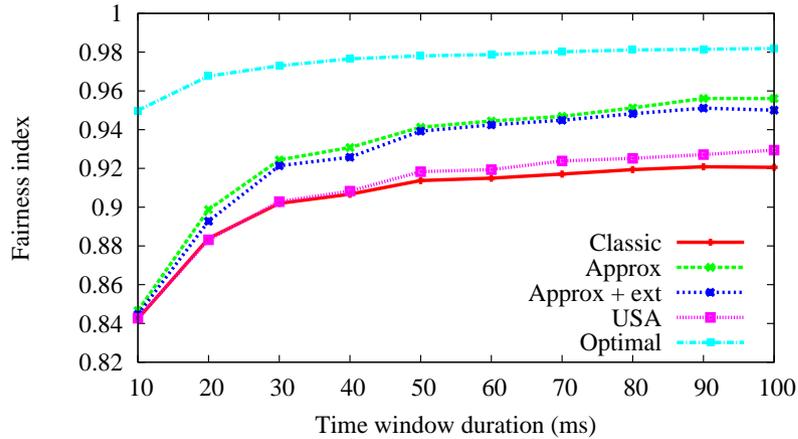


Figure 6. Fairness with PF.

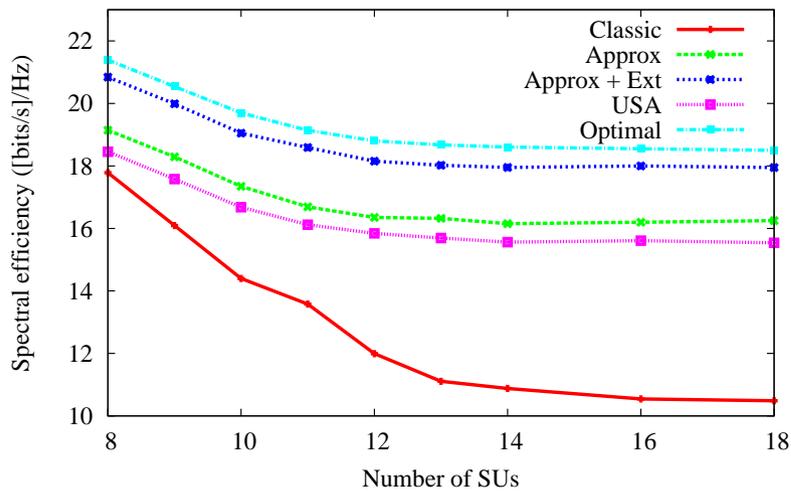


Figure 7. Spectral efficiency as a function of the number of SUs (with PF).

provides the highest capacity while achieving a very good fairness compared to other sub-optimal solutions is because the adapted PF policy considers (1) the single MCS constraint and (2) the TBS table to compute the rates dynamically in cooperation with our algorithm.

Fig. 7 depicts the average spectral efficiency as a function of the number of SUs. Again, our three-steps algorithm allocates the SUs in a more effective way compared to the other suboptimal algorithms. In this figure, the two curves are decreasing which could be surprising since the utility functions are superadditive. However, when the number of resources is higher, the probability of allocating a low quality SU is also higher. As a consequence, the single MCS used by each user is globally less effective compared to a case where only few SUs are available. However, we note that our algorithm limits this effect and remains close to the optimal spectral efficiency (with respect to the PF policy).

Scalability of our solution. To show that our algorithm can handle the highest system loads, we evaluated the performances with 50 users (under the same mobility assumptions as previously). Furthermore, the maximum number of available SUs was then set to 26 which is close to the maximum in an LTE-B network. In this scenario, we did not compare our solution with the optimal algorithm since it is not any more possible to compute an optimal solution in a reasonable time.

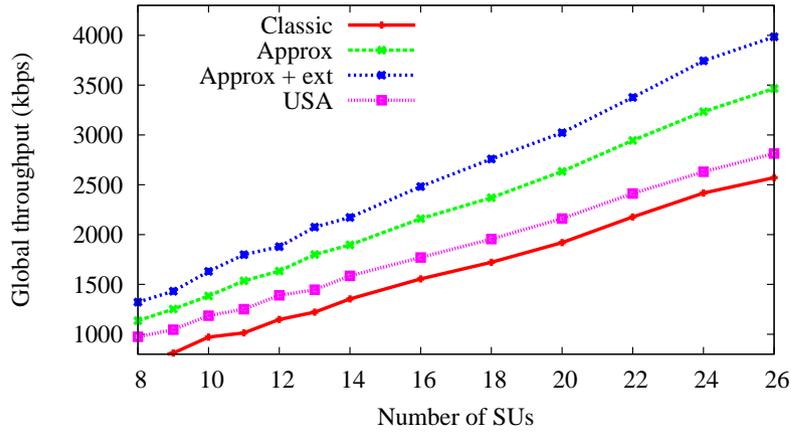


Figure 8. Capacity with PF (with 50 users and up to 26 SUs).

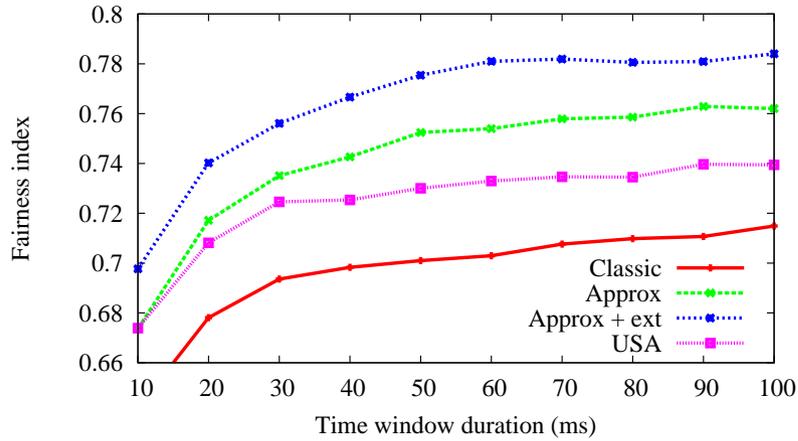


Figure 9. Fairness with PF (with 50 users and 20 SUs).

Fig. 8 compares our approximation algorithm with the other resource allocation algorithms in terms of average capacity as a function of the number of available SUs. Fig. 9 shows the fairness index as a function of Δt in a system composed of 20 SUs.

The trends of the curves are similar to what we obtained in the 10 users scenarii studied above (Scenarii 1 and 2). As expected, the gap between our algorithm (Approx + ext) and the other solutions is continuously increasing. Compared with the small instance of 10 users, we note that the resulting total throughput is higher for each algorithm, which is explained by a better use of the multi-user diversity. In addition, the fairness is lower because it is more difficult to be totally fair in a system with a large number of users.

6. CONCLUSION

In this paper, we studied a major issue in cellular networks: the resource allocation problem under the real and practical constraints of the LTE-B system. We first show that the problem is NP-Hard and therefore cannot be solved in a reasonable time. Hence, we proposed a generic approximation algorithm that tackles the LTE-B DL FDPS problem. Using system

level simulations, we showed that our algorithm achieves a high fairness level while achieving the highest data rate. Our study shows the importance of the LTE specificities resulting from the standard on resource allocation.

REFERENCES

1. Signal O. Lte coverage worldmap Nov 2015. URL <http://opensignal.com/>.
2. Solutions N. Lte deployment map Jan 2015. URL <http://ltemaps.org/home/>.
3. 3GPP. Physical layer procedures (Release 13). *TS 36.213*, 3GPP 2016. URL <http://www.3gpp.org/>.
4. Pokhariyal A, Kolding T, Mogensen P. Performance of downlink frequency domain packet scheduling for the utran long term evolution. *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, 2006; 1–5, doi:10.1109/PIMRC.2006.254113.
5. Dahlman E, Parkvall S, Skold J. *4G: LTE/LTE-Advanced for Mobile Broadband: LTE/LTE-Advanced for Mobile Broadband*. Elsevier.
6. Capozzi F, Piro G, Grieco LA, Boggia G, Camarda P. Downlink packet scheduling in lte cellular networks: Key design issues and a survey. *IEEE Commun. Surveys and Tutorials* April 2013; **15**(2):678–700. Doi:10.1109/SURV.2012.060912.00100.
7. Lee SB, Pefkianakis I, Choudhury S, Xu S, Lu S. Exploiting spatial, frequency, and multiuser diversity in 3gpp lte cellular networks. *Mobile Computing, IEEE Transactions on* Nov 2012; **11**(11):1652–1665, doi:10.1109/TMC.2011.206.
8. Niafar S, Huang Z, Tsang D. An optimal standard-compliant mimo scheduler for lte downlink. *Wireless Communications, IEEE Transactions on* May 2014; **13**(5):2412–2421, doi:10.1109/TWC.2013.031314.130723.
9. Ahmed H, Jagannathan K, Bhashyam S. Queue-aware optimal resource allocation for the lte downlink Dec 2013; :4122–4128.
10. Carpin M, Zanella A, Rasool J, Mahmood K, Grondalen O, Osterbo O. A performance comparison of lte downlink scheduling algorithms in time and frequency domains. *Communications (ICC), 2015 IEEE International Conference on*, 2015; 3173–3179, doi:10.1109/ICC.2015.7248812.
11. Lee SB, Choudhury S, Khoshnevis A, Xu S, Lu S. Downlink mimo with frequency-domain packet scheduling for 3gpp lte. *INFOCOM 2009, IEEE*, 2009; 1269–1277, doi:10.1109/INFCOM.2009.5062041.
12. Xiao X, Tao X, Lu J. Energy-efficient resource allocation in lte-based mimo-ofdma systems with user rate constraints. *Vehicular Technology, IEEE Transactions on* Jan 2015; **64**(1):185–197, doi:10.1109/TVT.2014.2319078.
13. Xiong K, Fan P, Lu Y, Letaief KB. Energy efficiency with proportional rate fairness in multi-relay OFDM networks. *CoRR* 2015; **abs/1511.07566**. URL <http://arxiv.org/abs/1511.07566>.
14. Abdel-Hadi A, Clancy C. A utility proportional fairness approach for resource allocation in 4g-lte. *Computing, Networking and Communications (ICNC), 2014 International Conference on*, 2014; 1034–1040, doi:10.1109/ICCNC.2014.6785480.
15. Shajaiah H, Abdel-Hadi A, Clancy C. Utility proportional fairness resource allocation with carrier aggregation in 4g-lte. *MILCOM 2013 - 2013 IEEE Military Communications Conference*, 2013; 412–417, doi:10.1109/MILCOM.2013.77.
16. Kwan R, Leung C, Zhang J. Multiuser scheduling on the downlink of an lte cellular system. *Rec. Lett. Commun.* Jan 2008; :3:1–3:4doi:10.1155/2008/323048.

17. Zhang H, Prasad N, Rangarajan S. Mimo downlink scheduling in lte systems. *INFOCOM, 2012 Proceedings IEEE*, 2012; 2936–2940, doi:10.1109/INFCOM.2012.6195733.
18. Liao HS, Chen PY, Chen WT. An efficient downlink radio resource allocation with carrier aggregation in lte-advanced networks. *Mobile Computing, IEEE Transactions on* Oct 2014; **13**(10):2229–2239, doi:10.1109/TMC.2013.2297310.
19. Song G, Li Y. Cross-layer optimization for ofdm wireless networks-part i: theoretical framework. *Wireless Communications, IEEE Transactions on* March 2005; **4**(2):614–624, doi:10.1109/TWC.2004.843065.
20. Feige U, Izsak R. Welfare maximization and the supermodular degree. *Electronic Colloquium on Computational Complexity (ECCC)* ; :95.
21. Schrijver A. *Theory of Linear and Integer Programming*. John Wiley & Sons, Inc.: New York, NY, USA, 1986.
22. IBM ILOG CPLEX Optimizer.
url<http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/> Last 2010.
23. Papadimitriou CM. *Computational complexity*. Addison-Wesley: Reading, Massachusetts, 1994.
24. Calinescu G, Chekuri C, Pál M, Vondrák J. Maximizing a submodular set function subject to a matroid constraint. *Proceedings of the 12th International Conference on Integer Programming and Combinatorial Optimization, IPCO '07*, Springer-Verlag: Berlin, Heidelberg, 2007; 182–196.
25. Feige U. On maximizing welfare when utility functions are subadditive. *SIAM Journal on Computing* 2009; **39**(1):122–142, doi:10.1137/070680977.
26. Andrews M. A survey of scheduling theory in wireless data networks. *Wireless Communications, The IMA Volumes in Mathematics and its Applications*, vol. 143, Agrawal P, Fleming P, Zhang L, Andrews D, Yin G (eds.). Springer New York, 2007; 1–17, doi:10.1007/978-0-387-48945-2_1.
27. Holzman R, Kfir-Dahav N, Monderer D, Tennenholtz M. Bundling equilibrium in combinatorial auctions. *Games and Economic Behavior* 2004; **47**(1):104 – 123, doi:[http://dx.doi.org/10.1016/S0899-8256\(03\)00184-2](http://dx.doi.org/10.1016/S0899-8256(03)00184-2).
28. Mirrokni V, Schapira M, Vondrak J. Tight information-theoretic lower bounds for welfare maximization in combinatorial auctions. *Proceedings of the 9th ACM Conference on Electronic Commerce*.
29. RiverBed. Opnet modeler with wireless suite Jun 2013. URL <http://www.riverbed.com/>.
30. Blankenship Y, Sartori P, Classon B, Desai V, Baum K. Link error prediction methods for multicarrier systems. *IEEE Vehicular Technology Conference*, vol. 6, 2004; 4175–4179, doi:10.1109/VETECF.2004.1404865.
31. Chen FT, Tao GL. A novel mcs selection criterion for supporting amc in lte system. *International Conference on Computer Application and System Modeling (ICCASM)*, vol. 6, 2010; 598–603, doi:10.1109/ICCASM.2010.5620840.
32. Fan J, Yin Q, Li G, Peng B, Zhu X. Mcs selection for throughput improvement in downlink lte systems. *Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)*, Maui, Hawaii, 2011; 1–5, doi:10.1109/ICCCN.2011.6005743.
33. Ragaleux A, Baey S, Gueguen C. Adaptive and generic scheduling scheme for lte/lte-a mobile networks. *Wireless Networks* 2015; :1–19doi:10.1007/s11276-015-1131-9. URL <http://dx.doi.org/10.1007/s11276-015-1131-9>.