Diez, L., García-Saavedra, A., Valls, V., Li, X., Costa-Pérez, X. y Agüero, R. (2018). LaSR: A Supple Multi-Connectivity Scheduler for Multi-RAT OFDMA Systems. *IEEE Transaction on Mobile Computing.*

DOI: https://doi.org/10.1109/TMC.2018.2876847

LaSR: A Supple Multi-Connectivity Scheduler for Multi-RAT OFDMA Systems

Luis Diez, Andres Garcia-Saavedra, Víctor Valls, Xi Li, Xavier Costa-Perez, Ramón Agüero

Abstract—Network densification over space and spectrum is expected to be key to enabling the requirements of next generation mobile systems. The pitfall is that radio resource allocation becomes substantially more complex. In this paper we propose LaSR, a practical multi-connectivity scheduler for OFDMA-based multi-RAT systems. LaSR makes optimal discrete control actions by solving a sequence of simple optimization problems that do not require prior information of traffic patterns. In marked contrast to previous work, the flexibility of our approach allows us to construct scheduling policies that achieve a good balance between system cost and utility satisfaction, while jointly operate across heterogeneous RATs, accommodate real-system requirements, and guarantee system stability. Examples of system requirements considered in this paper include (but are not limited to): constraints on how scheduling data can be encoded onto signaling protocols (e.g. LTE's DCI), delays when turning on/off radio units, or on/off cycles when using unlicensed spectrum. We evaluate our scheduler via a thorough simulation campaign in a variety of scenarios with e.g. mobile users, RATs using unlicensed spectrum (using a duty cycle access mechanism), imperfect queue state information, and constrained signaling protocol.

Index Terms—Radio resource scheduling, multi-connectivity, carrier aggregation

1 Introduction

Network densification is well-recognized as a key means to take on the challenge of supporting a thousand-fold increase in traffic demand in the next generation of mobile systems [1]. In turn, network densification involves both *spatial densification* (packing more radio access points per unit area) and *spectrum densification* (aggregating potentially non-contiguous radio bands) [2].

A cost-efficient way of accomplishing spatial densification is to deploy an "army" of low-power low-cost radio access technologies (RATs) such as small-cells (see Fig. 1). The advantages of this approach are well known, namely (i) distance between users and RATs is shortened, thus increasing the quality of the wireless links; and (ii) (small) RATs can implement more aggressive energy-saving features, lowering the operational costs of the infrastructure. On the downside, however, the load that each individual RAT has to manage becomes highly volatile and unpredictable.[1] Hence, a cost-efficient dense deployment requires a flexible and adaptive control of radio resources. For instance, a network operator may want to distribute low-power load across fewer RATs and/or use only inexpensive unlicensed bands as long as the demand is satisfied in order to save costs and cause low interference. However, when the load increases, the network controller needs to adapt very quickly (e.g. activating RATs to offload traffic during peak hours).

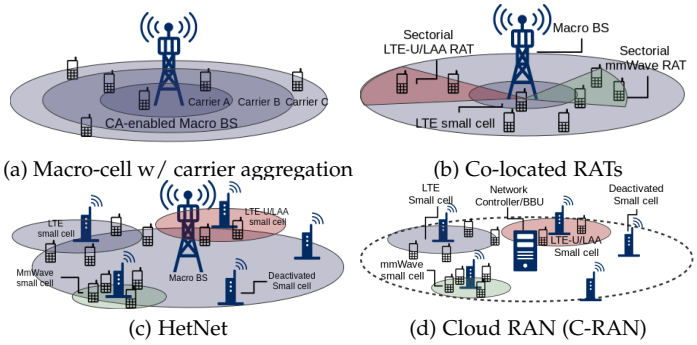


Fig. 1. Deployments relevant for 5G and beyond [6].

Regarding spectrum densification, multi-connectivity between single users and multiple RATs is attracting a lot of interest to 5G RAN architects, who are in the hunt for larger chunks of spectrum [7]. With multi-connectivity we can extend the amount of bandwidth by aggregating non-contiguous bands, e.g., sub-6GHz, ISM bands, mmWave or TV white spaces (see Fig. 1)—unified under a common OFDM-based air interface, namely 3GPP's 5G New Radio (NR) [8]. Although multi-connectivity can be implemented at different layers of the stack (TCP/IP, PDCP or MAC [7]) in this paper we consider MAC layer aggregation because it allows much finer granularity and it is a natural evolution of legacy Carrier Aggregation (CA)—introduced by 3GPP in LTE Release 10 specification [9].

As a consequence of the above, heterogeneity across RATs and resource demand volatility makes radio resource scheduling substantially more complex. In this paper we design LaSR (Lagrange approximation Supple Radio controller), a practical multi-connectivity scheduler that assigns radio resources to mobile users in an OFDMA-based multi-RAT system, having the following main features:

- *A. Garcia-Saavedra, X. Li and X. Costa-Perez are with NEC Laboratories Europe, Germany.*
  E-mail: {andres.garcia.saavedra, xi.li, xavier.costa}@neclab.eu
- *V. Valls is with Trinity College Dublin, Ireland.*
  E-mail: victor.valls@tcd.ie
- *L. Diez and R. Agüero are with University of Cantabria, Spain.*
  E-mail: {ldiez, ramon}@tlmat.unican.es

- *The authors from University of Cantabria would like to thank the Spanish Government (Ministerio de Economía y Competitividad, Fondo Europeo de Desarrollo Regional, FEDER) for its support through the project ADVICE (TEC2015-71329-C2-1-R).*

1. User traffic is highly variable, as evidenced in a plethora of literature [3]–[5], but macro-cells compensate this volatility by aggregating multiple flows. This leverage fades in dense contexts because individual (low-power) RATs handle fewer flows [1].

- LaSR operates using instantaneous state information of network queues with no assumptions on the stochastic properties of user traffic or available bandwidth, adapts quickly to changing conditions, and guarantees stability even with imperfect information;
- LaSR stabilizes all network queues (maximizes throughput) while balancing network cost and a proportionally-fair allocation of individual utilities;
- The *suppleness* of LaSR is the main advantage with respect to state of the art techniques because, differently to prior work, LaSR allows us to satisfy practical real-system constraints, e.g. delays when switching on/off RATs, discrete and heterogeneous ranges of modulations and resource blocks, constrained encoding of scheduling information into signaling protocols (e.g. LTE uses a Downlink Control Indicator (DCI) that trades off granularity with reduced overhead), channel unavailability due to coexistence operation in unlicensed bands, etcetera.

The rest of the paper is organized as follows. In §2 we revise related work and discuss the advantages of LaSR. We introduce a mathematical model of our system and optimization problem in §3. In §4 we design an algorithm that iteratively exploits information from the convex relaxation of our problem so solve it efficiently. In §5 we present a thorough performance evaluation of the proposed scheme. Finally, §6 includes some final remarks.

2 Related Work & Discussion

Related Work. Resource allocation in OFDM-based systems has been widely studied [10], [11]. The assumption of an infinitely backlogged model is conventional in the related literature. Under this assumption, many works study bounds on the total utility a system can achieve, e.g. [12], [13], including throughput, delay, energy and fairness considerations. Only recently, a few works consider carrier aggregation [14]–[18]. In [14], [15], the authors focus on allocating rates to maximize the network utility as a proportionally fair allocation of utilities of different traffic classes; the latter supporting user discrimination. In both, users are allocated resources sequentially, from a sorted list of carriers, i.e., without load balancing. This work was extended in [16] to minimize user costs (e.g. leasing, energy consumption), and energy efficiency is considered in [17]. In terms of complexity, all these works require solving a sequence of convex programs at every slot (mild complexity). Finally, in [18], the authors address the rate allocation problem using a game-theoretic approach, suitable for a distributed setting, which requires a number of operations that scale exponentially with the number of users and RATs. Although all these works offer a very valuable theoretical analysis of the performance boundaries of carrier aggregation technology, they are not applicable in practice because, in contrast to ours, they either (i) operate only in the capacity boundary (very high load), (ii) do not let RATs be deactivated to reduce costs in low-, mid-load regimes, and/or (iii) do not consider practical issues of real systems like e.g. how to map rates to discrete sets of modulations/resource blocks or signaling constraints. Remarkably, as we will show later, LaSR satisfies all these requirements with similar mild complexity, i.e., by solving a sequence of convex programs.

TABLE 1
Related Work with CA support

	LaSR	[14]	[15]	[16]	[17]	[18]	[19]	[20]	[21]
Performance guarantees	✓	✓	✓	✓	✓	✓			
Network cost	✓			✓	✓			✓	
QoS diff.	✓	✓	✓	✓		✓			
All load regimes	✓						✓	✓	✓
Traffic pattern oblivious	✓								✓
Heterogeneous RATs	✓								✓‡
Flexibility†	✓								✓‡
Complexity	M	M	M	M	M	M	L	L	L

† to accommodate system constraints while preserving optimality
‡ only partially
M: Mild
L: Low

There exists some research on carrier aggregation relaxing channel assumptions. For instance, a simple proportional fair scheduler with no power control is analyzed in [19] and energy efficiency in [20]. Both approaches have low complexity (involving simple sorting operations). However, a Poisson (elastic) traffic model and homogeneity across RATs is assumed. Finally, the authors of [21] propose a backlog-based heuristic algorithm, comparable to our work in its applicability to practical systems, but with no performance guarantees, and assuming homogeneity across RATs. In comparison, LaSR achieves provably optimal performance without taking assumptions on the traffic model, accommodating scheduling decisions to system constraints, such as delays when turning on/off RATs, signaling limitations, heterogeneous resources, or imperfect information with mild complexity (involving a sequence of convex problems) and, importantly, allows us to trade-off computational burden for (slower) reaction time without compromising optimality. Table 1 summarizes the above comparison.

In a broader context (neglecting carrier aggregation), much literature addresses the scheduling problem, yet making limiting assumptions regarding traffic stochastic properties (e.g. [22], [23]). A few works relax those assumptions, e.g., [24], [25] propose a Markov Decision Problem (MDP) model, which renders a very practical scheduler but optimality suffers of the *curse-of-dimensionality* and the authors have to settle with approximate solutions. Another example is [26], which proposes a simple suboptimal greedy algorithm to solve a binary integer program. There exist some scheduling policies known to be throughput-optimal, namely: max-weight [27], exp-rule [28], and log-rule [29]. Max-weight policies select to transmit, at each time slot, the subset of queues with the maximum *weight*, typically defined in OFDM systems as the current backlog or the product of the maximum feasible service rate (i.e. modulation level) and the backlog. Exp-rule aims to minimize the exponential decay rate of delay distribution tail of the worst user. Log-rule uses a Markov chain to obtain the queue state transition probabilities, which are used to minimize the average queue lengths (and thus delay). Recently, the authors of [30] adapted the heavy-ball technique [31] to

the wireless network utility maximization problem with the objective of reducing convergence times and user delay.

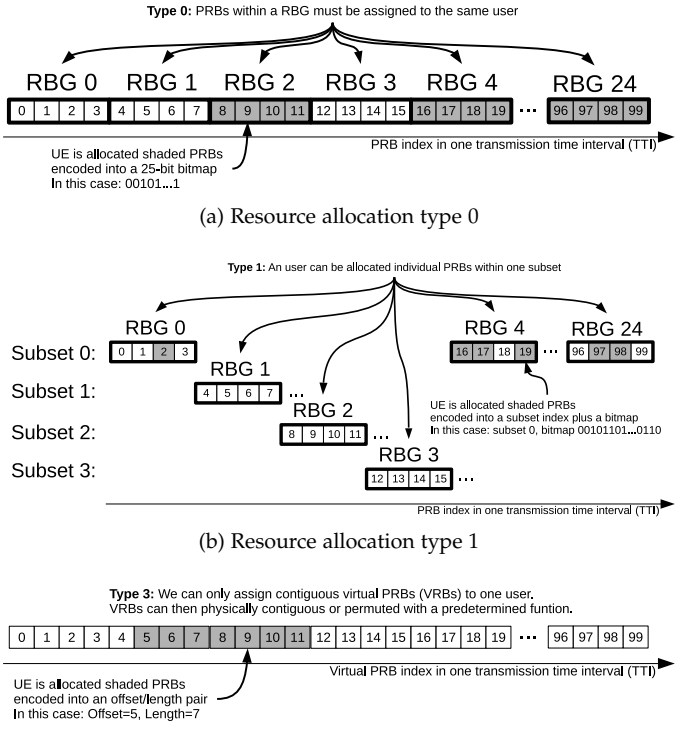
Discussion. These throughput-optimal techniques, however, do not provide much flexibility when designing feasible scheduling policies and they are thus hard to implement when facing practical constraints. For example:

(i) A desirable feature in multi-RAT systems is the ability to deactivate secondary RATs as much time as possible, e.g. during low-load regimes, to reduce operational costs, and turn them back on as soon as possible when load builds up. However, real hardware needs some time to switch back and forth between operational and sleep states. Therefore, our scheduler needs to guarantee that resources are not assigned to RATs while they are in sleep mode or are being activated;

(ii) RATs can be heterogeneous in terms of available number of physical resource blocks (PRBs) per time×bandwidth unit, available modulations and/or duty cycles when using unlicensed bands. A cross-RAT controller should manage this heterogeneity;

(iii) Operators may be willing to trade off delay performance of some (e.g. best effort) users to save operational costs and/or reduce interference to neighboring systems. In such cases, an advantageous policy would be favoring low modulation levels or that extending the amount of time a RAT is off as long as mean demand rate is met. Though perhaps counter-intuitive, the rationale is that low modulation levels let RATs use lower transmission power and we could thus trade off *local* spectral efficiency for higher spatial diversity, which is of great importance in dense scenarios because individual RATs manage fewer users;

(iv) Real systems may impose constraints in the way actions are taken. For instance, LTE uses a Downlink Control Indicator (DCI) in the PDCCH (Physical Downlink Control Channel) that carries information regarding which resource blocks carry data for which user. Obviously, there is a limitation in the way this information is encoded in signaling protocols in order to trade off granularity with reduced overhead. Let us illustrate this by explaining how a resource allocation can be encoded into LTE's DCI. As depicted in Fig. 2, resource allocation in LTE can be done in 3 ways: type 0, 1 and 2. In type 0, consecutive PRBs are grouped into RBGs (e.g. in a 20 MHz RAT, each RBG corresponds to 4 PRBs). In this way, a scheduler is constrained to take the same action for each of the PRBs in one group. In type 1, RBGs are grouped into subsets (via a standard modulo relationship) and users are allocated individual PRBs within one subset. In this way, a scheduler is constrained to a subset of PRBs per user. In type 2, we can allocate any number of virtually continuous PRBs via an offset/length pair. Then, these PRBs can either be physically continuous or distributed by a permutation function specified in the standard. In either case, a scheduler will be also constrained to assigning PRBs in groups. See [32] for more details. In summary, a practical scheduler should guarantee that this type of constraints—which will always exist in real protocols—are respected in order to extend stability guarantees to real implementations.



(a) Resource allocation type 0

(b) Resource allocation type 1

(c) Resource allocation type 2

Fig. 2. DCI Resource Allocation types in a 20-MHz LTE RAT (100 PRBs available per allocation period or transmission time interval (TTI)).

Unfortunately, these practical requirements are inherently hard to meet with the aforementioned approaches. Like some prior work [33]–[37], we exploit the connection between queue states and Lagrange multipliers. However, all these approaches rely on continuous variables and/or greedily select actions in a myopic manner, i.e., based only on the current state. As a result, they cannot take into account that there might be a subset of actions that are not allowed or are not preferred at some time. Thus, in order to comply with real-life constraints, *the sample trajectory of actions*, rather than simply the average, shall be considered. There exists some literature that considers "constrained" actions. For instance, [35] shows that a myopic policy cannot perform optimally when there are reconfiguration delays, and proposes a solution for that specific case. However, it is not clear how this approach can be used to encompass more complex constraints like the ones considered in this paper.

Recently, [38] showed that the choice of a discrete control action can be decoupled from a specific choice of subgradient. This is of paramount importance for our work because it gives us a lot of flexibility to design policies that accommodate practical constraints of real systems without compromising the underlying convex updates. Based on this fundamental idea, in this paper we design LaSR, a multi-connectivity scheduler for multi-RAT systems that is sufficiently supple to operate optimally in a variety of scenarios (like the ones in Fig. 1). Yet, it does not take assumptions on the stochastic properties of the system (arrival of data, mobility, etc.), nor does it make simplifications on the constraints of the underlying system (e.g., discrete sets of modulations and PRBs, signaling overhead, imperfect—noisy or delayed—backlog information, reconfiguration de-

lays, etc.) and can balance operator and users preferences.

3 System Model

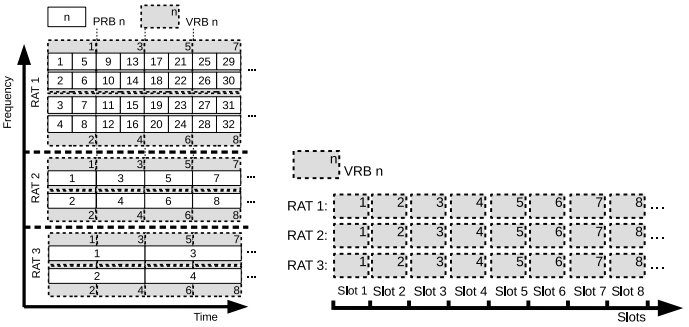
We introduce in this section a model based on a system of queues. Then, based on this model, we formalize the description of our problem as an optimization problem.

3.1 Notation

We use conventional notation. We let $\mathbb{R}$ and $\mathbb{Z}$ denote the set of real and integer numbers. We use $\mathbb{R}_+$, $\mathbb{R}^n$, and $\mathbb{R}^{n\times m}$ to represent the sets of non-negative real numbers, n-dimensional real vectors, and $m \times n$ real matrices, respectively. Vectors are usually in column form and written in bold font. Matrices are in upper-case font. Subscripts represent an element in a vector and superscripts elements in a sequence. For instance, $\langle \boldsymbol{x}^{(t)} \rangle$ is a sequence of vectors with $\boldsymbol{x}^{(t)} = [x_1^{(t)}, \ldots, x_n^{(t)}]^T$ being a vector from $\mathbb{R}^n$. In turn, $x_i^{(t)}$ is the i'th component of the t'th vector in the sequence. Superscript T represents the transpose operator, we use $\mathbf{1}$ or $\mathbf{0}$ to indicate a vector where all elements are 1 or 0, respectively, and $\boldsymbol{x} \preceq \boldsymbol{y}$ indicates that $x_i \leq y_i, \forall i$. $\|\boldsymbol{x}\|_2$ represents the 2-norm or Euclidean norm of $\boldsymbol{x}$ and $\|\boldsymbol{x}\|_\infty$ its maximum norm ($\max_i |x_i|$). Finally, $[\cdot]^+$ denotes the projection of a vector onto the non-negative orthant, i.e., $[\boldsymbol{x}]^+ = [\max\{0, x_1\}, \ldots, \max\{0, x_n\}], x \in \mathbb{R}^n$.

3.2 Model

We consider a multi-RAT system comprised of $\mathcal{R} = \{1, \ldots, R\}$ RATs and $\mathcal{N} = \{1, \ldots, N\}$ mobile users. RATs can be co-located or distributed in a C-RAN architecture (perfect information) or forming a HetNet structure (imperfect information). Without loss of generality, we assume that each user is mapped to one traffic class or QoS class identifier (QCI). The extension to a general case can be done by adding virtual users and aggregating traffic of the same type from multiple users into one virtual user. All RATs can be deactivated except one (primary RAT), in order to guarantee an available control channel at all times. Following 5G New Radio (NR) design, we assume all RATs are based on OFDM [8] so that we can allocate physical resource blocks (PRBs) from a pool available at each RAT. Each PRB can be modulated with a modulation level from the discrete set $\mathcal{M}_{r,n} = \{m_{r,n,1}, \ldots, m_{r,n,M}\}$, $\forall r \in \mathcal{R}, \forall n \in \mathcal{N}$, where $m_{r,n,M}$ is the highest modulation level that can be used in the physical link between RAT r and user n. We can easily compute $m_{r,n,M}$ using standard models by exploiting channel state information (CSI) of the wireless channel between each RAT and user (e.g. using CQI (Channel Quality Indicator) reports fed back by users readily available in LTE systems). The accuracy of this information or how to best select $m_{r,n,M}$ based on this (possibly imperfect and quantized) information is out of the scope of this paper; we refer the reader to [39] and references therein to find literature on the topic. Importantly, however, the algorithm we design in §4 does consider that (higher) modulation levels could be unavailable at some times due to fading, i.e., system stability is preserved. Our goal is to design a network controller that jointly assigns PRBs, modulation levels and RAT(s) to users such that the demand is satisfied (system is stable) while maintaining a good balance between system cost and QoS satisfaction.



(a) VRB mapping (b) Slotted system

Fig. 3. Three RATs with capacity 32, 8, and 4 PRBs, respectively, per time×bandwidth. RATs have equal number of VRBs per bandwidth×time unit (left). Our controller operates in slots, each slot corresponding to 1 VRB per RAT (right).

Independently of the (possibly different) physical division of spectrum/time resources into PRBs of each RAT, we divide them into an equal number of virtual resource blocks (VRB) per bandwidth×time unit, as depicted in Fig. 3a. The actual number of VRB per unit of resource is an implementation choice that may depend on the computational capacity available (more VRBs/resource provides more granularity but requires faster computational operation). Note that this may cause that a VRB contains more than one PRB in some RATs, e.g., if a PRB is "smaller" in time×bandwidth than the chosen VRB like RAT 1 in Fig. 3a, or even fractions of PRBs, e.g., if one RAT has more assignable PRBs than VRBs like RAT 3 in Fig. 3a. However, this approach allows us to *homogenize* the resources of potentially heterogeneous RATs, which substantially simplifies our model. We note that, though this simplification may render suboptimal decisions if not handled carefully, our scheduler maps VRBs into (heterogeneous) per-RAT PRBs *with no loss in optimality*. This is explained in in §4.2. Although we focus on the downlink case hereafter, our model also applies to the uplink case (in fact we evaluate an uplink scenario—with imperfect information—later on).

As illustrated in Fig. 4, we model this system as a network of $2N$ queues (an incoming queue and outgoing queue per traffic type) and $L = \sum_{n\in\mathcal{N}} \sum_{r\in\mathcal{R}} |\mathcal{M}_{r,n}|$ links (different ways of transmitting data between RATs and users). Our controller operates in slots $t = 1, 2, \ldots$, each slot containing one VRB per RAT, as shown in Fig. 3b. We then model the dynamics of the queues as

$$\boldsymbol{Q}^{(t+1)} = \left[\boldsymbol{Q}^{(t)} + \boldsymbol{\Delta}^{(t)}\right]^+, \quad t = 1, 2, \ldots$$

where $\boldsymbol{Q}^{(t)} \in \mathbb{Z}_+^{2N}$ is a column vector bookkeeping the state of all queues in the system at slot t, i.e. $\boldsymbol{Q}^{(t)} = [Q_1^{(t)}, Q_2^{(t)}, \ldots, Q_{2N}^{(t)}]^T$ and $\boldsymbol{\Delta}^{(t)} \in \mathbb{Z}^{2N}$ is a column vector containing the queues net increments/decrements of data units in slot t. We now let incidence matrix $A \in \mathbb{Z}^{2N\times L}$ represent the connections between queues, so that element $A_{i,j}$ represents the amount of *data units per VRB* departing from (if negative) or arriving to (if positive) queue i when activating link j. Note that the amount of bits transported on each VRB depends on the modulation scheme used in link j at one slot. Fig. 5 illustrates an example with 2 users,

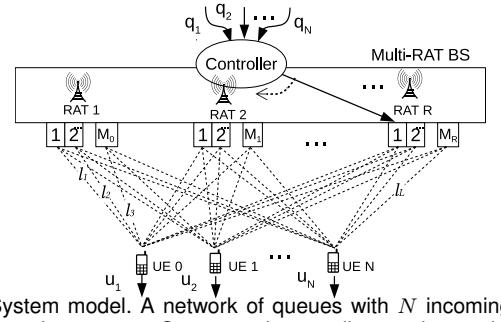


Fig. 4. System model. A network of queues with N incoming queues and N outgoing queues. Our network controller needs to select a RAT and a modulation index (i.e. a link connecting queues) to deliver data between each pair of queues $\{Q_i, Q_{N+i}\}$.

2 RATs and 2 modulation indexes available for each RAT and user, and the incidence matrix A equal to

$$\begin{array}{c} \\ Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{array} \begin{array}{c} \begin{array}{cccccccc} l_1 & l_2 & l_3 & l_4 & l_5 & l_6 & l_7 & l_8 \\ (m_{1,1,1}) & (m_{1,1,2}) & (m_{2,1,1}) & (m_{2,1,2}) & (m_{1,2,1}) & (m_{1,2,2}) & (m_{2,2,1}) & (m_{2,2,2}) \end{array} \\ \begin{pmatrix} -1 & -2 & -10 & -20 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -2 & -10 & -20 \\ 1 & 2 & 10 & 20 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 10 & 20 \end{pmatrix} \end{array} \quad (1)$$

represents the connections between queues in this example.

At each slot, the controller makes a scheduling decision and so it assigns data into VRBs (or not) by activating/deactivating links in our model. Precisely, let $Y \subseteq \{0,1\}^L$ collect all possible actions in the system and vector $\boldsymbol{y}^{(t)} \in Y$ represent the action taken at slot t, where the i'th element of the vector indicates whether link i is active (if equal to 1) or not (otherwise) in slot t. In this way, in each slot our controller assigns resources from the available RATs to users and selects a modulation level for each active pair of RAT/user. Hence, at each slot we have the update

$$\boldsymbol{Q}^{(t+1)} = \left[\boldsymbol{Q}^{(t)} + A\boldsymbol{y}^{(t)} + \boldsymbol{b}^{(t)}\right]^+, \quad (2)$$

where $A\boldsymbol{y}^{(t)}$ captures the outcome of taking decision $\boldsymbol{y}^{(t)} \in Y$ at this time, and $\boldsymbol{b}^{(t)} \in \mathbb{Z}^{2N}$ is a vector with the net amount of bits that enter/leave each queue in the system at one slot. In practice $b_i = 0, \ \forall i > N$ since these queues are the destination of the data. In our simple example, $\boldsymbol{y}^{(t)} = [1,0,0,1,0,0,0,0]^T$ causes the following updates

$$Q_1^{(t+1)} = \left[Q_1^{(t)} - 21 + b_0^{(t)}\right]^+ \qquad Q_2^{(t+1)} = \left[Q_2^{(t)} + b_1^{(t)}\right]^+$$
$$Q_3^{(t+1)} = \left[Q_3^{(t)} + 21\right]^+ \qquad Q_4^{(t+1)} = \left[Q_4^{(t)}\right]^+.$$

That is, in slot $t+1$, user 1 is assigned a VRB from RAT 1 modulated with the lowest modulation level, and also a VRB from RAT 2 modulated with the highest modulation.

3.3 Problem Formulation and Approach

Our goal is to design a scheduling policy (i.e. a sequence of control actions) that (i) satisfies the queue dynamics given in Eq. (2), (ii) maximizes throughput, and that (iii) minimizes a utility function of the average of the actions, *while meeting practical constraints* such as discrete modulation sets, RAT (de)activation delays or signaling limitations.

Let us first introduce the following definitions.

Definition 1 (Scheduling policy). *A sequence* $\langle \boldsymbol{y}^{(t)} \mid \boldsymbol{y}^{(t)} \in Y, t \geq 0 \rangle$ *describes a* **scheduling policy** π.

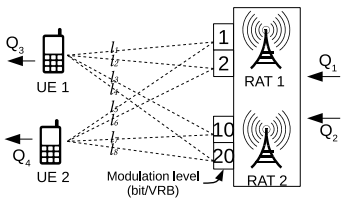


Fig. 5. Example with 2 RATS, 2 users, 2 available modulations per RAT and user. Queues $\boldsymbol{Q_1}/\boldsymbol{Q_3}$ and $\boldsymbol{Q_2}/\boldsymbol{Q_4}$ are respectively incoming/outgoing queues for user 1 and 2. Links $\{l_1 \dots l_8\}$ connect all queues in the system according to incidence matrix shown in Eq. (1), i.e., allowing different amount of bits per slot (different modulation levels), i.e., $m_{1,1,1} = m_{1,2,1} = 1$ bit/slot and $m_{1,1,2} = m_{1,2,2} = 2$ bits/slot (from RAT 1 to user 1 and 2), and $m_{2,1,1} = m_{2,2,1} = 10$ bits/slot and $m_{2,1,2} = m_{2,2,2} = 20$ bits/slot (from RAT 2 to user 1 and 2).

Definition 2 (Admissible policy). *Let* $\boldsymbol{\mathcal{A}}^{(t)}$ *define a set containing all the available actions* $\boldsymbol{\mathcal{A}}^{(t)} \subseteq Y$ *at time* t *(some actions may not be available at some slot due to e.g. constraints on the signaling information or RAT unavailability). Then, an* **admissible scheduling policy** π *is such that* $\pi = \langle \boldsymbol{y}^{(t)} \mid \boldsymbol{y}^{(t)} \in \boldsymbol{\mathcal{A}}^{(t)}, t \geq 0 \rangle$. *Finally,* **let superset** Π **collect all admissible scheduling policies.**

Then, our optimization problem can be written as

Problem 1 (LaSR Problem)**.**

$$\underset{\pi \in \Pi}{\text{minimize}} \qquad f\left(\frac{1}{T}\sum_{\substack{t=1 \\ \boldsymbol{y} \in \pi}}^{T} \boldsymbol{y}^{(t)}\right) \quad (3)$$

$$\text{subject to} \qquad \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\boldsymbol{Q}^{(t)}\right] \prec \infty \quad (4)$$

where constraint (4) corresponds to the strong stability requirement used in max-weight approaches, i.e., the demand for resources will be satisfied asymptotically. Note that a direct consequence of this constraint is that the delay experienced by users will be also bounded [40]. Hence, our problem consists on finding a sequence of discrete actions $\langle \boldsymbol{y}^{(t)} \rangle$ (an admissible policy $\pi \in \Pi$) that minimizes an *average* cost function $f(\cdot)$, subject to the stability constraint.

It is important to emphasize that Problem 1 is non-convex, since it requires selecting a sequence of actions from a discrete set Y. There are several approaches in the literature that consider the minimization of a utility function subjected to constraint (4). One of the most popular ones is the drift-plus-penalty algorithm (see for instance [36]). This is a greedy approach that consists of selecting a control action in each time slot with update

$$\boldsymbol{y}^{(t)} \in \arg\min_{\boldsymbol{y} \in Y} \quad VU(\boldsymbol{y}) + (\boldsymbol{Q}^{(t+1)})^T A\boldsymbol{y} \quad (5)$$

where $U : \mathbf{R}^n \to \mathbf{R}$ is a utility function, and $V \geq 0$ a tuning parameter that trades-off utility accuracy and delay.[2] One of the drawbacks of this approach is that a control action is selected in a myopic manner, i.e., based only on the current state of the system. As a result, it cannot take into account that there might be a subset of (non-admissible) actions

2. When parameter V is large, the update in Eq. (5) gives preference to minimizing the utility rather than emptying the queues with more packets. As a result, queues backlog increases (so the delay) and are not emptied until they become sufficiently large.

that are not allowed at a certain instant of time, or are not preferred because of a metric of interest, for instance, a RAT may be unavailable at some slot due to the delay it takes for turning on, or an action may have to be repeated due to restrictions on the signaling protocol (see LTE DCI in §2). Thus, in order to comply with real-life constraints, *the sample path or trajectory of actions*, rather than simply the average, shall be considered. There are some approaches in the literature that consider "constrained" control actions. For instance, [35] shows that a myopic or greedy policy cannot solve problems that have the form of Problem 1 when there are reconfiguration delays, and proposes a solution for that specific case. However, it is not clear how the approach can be used to encompass more complex control constraints, like the ones considered in this paper.

The approach we adopt here is based on [38], which shows that to obtain an optimal policy it is enough to find a sequence of discrete actions that remains close to a "relaxed" or "convex" sequence.[3] The key idea is that, since we are interested in minimizing a function $f(\cdot)$ of an average, the average policy (or sequence) can be constructed in multiple ways. Namely, there is no need to select control actions greedily in each time slot, as long as the average $T^{-1}\sum_{t=1}^{T}\boldsymbol{y}^{(t)}$ minimizes $f(\cdot)$. This allows us to construct policies that provably solve Problem 1 *and are admissible*.

In the sequel, we introduce a relaxed convex formulation and present the main contribution of our paper: an algorithm that, by solving a sequence of convex problems, makes scheduling decisions that solve (non-convex) Problem 1 while accommodating practical system constraints.

4 Algorithm Design

We first introduce the convex formulation of the problem, then we show that Lagrange multipliers and queues are related, and finally we present our main contribution: an algorithm that obtains a sequence of control actions that solves Problem 1, while satisfying the system constraints.

4.1 Convex Formulation

Consider the following convex problem.

Problem 2 (Convex Optimization Problem)**.**

$$\underset{\boldsymbol{x}\in X}{\text{minimize}} \quad f(\boldsymbol{x}) \tag{6}$$

$$\text{subject to} \quad A\boldsymbol{x}+\bar{\boldsymbol{b}}\preceq 0 \tag{7}$$

where A is an incidence matrix as described in §3, $\bar{\boldsymbol{b}}\in\mathbb{R}^{2N}$, and X is a convex subset from $\mathrm{conv}(Y)$ where Y is the set of all possible actions. That is, a vector $\boldsymbol{x}\in X\subseteq\mathrm{conv}(Y)$ indicates the *fraction* of time each link should be scheduled with data by a policy π, and $f(\boldsymbol{x})$ is a cost derived from that allocation. Also, note that $\bar{\boldsymbol{b}}$ represents the (unknown) mean arrival/departure data rate at each node and so constraint (7) guarantees that all data arriving into the system is served (and so it relates to the stability constraint of Problem 1). We will make the usual assumption that $X^*:=\{\arg\min_{\boldsymbol{x}\in X} f(\boldsymbol{x}) \mid A\boldsymbol{x}+\bar{\boldsymbol{b}}\preceq 0\}$ is nonempty, and so Problem 2 is feasible.

We are interested in solving Problem 2 efficiently, and so we need to (i) ensure that objective function selected is convex (and so Problem 2); but also (ii) that the algorithm used does not require perfect knowledge of $\bar{\boldsymbol{b}}$. The first point is important because then we can use standard convex optimization methods, while the second one is because, as noted in §1, resource demand is hard to predict (if possible at all) in dense multi-RAT systems. We address these two points in more detail next.

4.1.1 Lagrange Relaxation and Dual Problem

We can relax the perfect knowledge of the constraints by applying Lagrange relaxation. In short, Lagrange relaxation allows us to formulate the dual problem, with which we can generate a sequence of primal variables $\langle\boldsymbol{x}^{(t)}\rangle$ that converges to the optimum or a point nearby without requiring to be feasible in each iteration. This is in marked contrast to other iterative methods, such as interior point or projected gradient, and so provides flexibility to select a sequence of actions or gradients.[4] This will be clear shortly, but we first introduce the Lagrange dual problem of Problem 2:

Problem 3 (Lagrange Dual Problem)**.**

$$\underset{\boldsymbol{\lambda}\succeq 0}{\text{maximize}} \quad q(\boldsymbol{\lambda}) \tag{8}$$

where $q(\boldsymbol{\lambda}):=\inf_{\boldsymbol{x}\in X}\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda})$ *with* $\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda})=f(\boldsymbol{x})+\boldsymbol{\lambda}^T(A\boldsymbol{x}+\bar{\boldsymbol{b}})$.

We note that solving Problem 3 is equivalent to solving Problem 2 when strong duality holds,[5] and that $q(\boldsymbol{\lambda})$ is concave [41]. Hence, $q(\boldsymbol{\lambda})$ can be maximized using the standard (sub)gradient ascent method with fixed step, i.e.,

$$\boldsymbol{\lambda}^{(t+1)}=\left[\boldsymbol{\lambda}^{(t)}+\alpha\left(A\boldsymbol{x}^{(t)}+\bar{\boldsymbol{b}}\right)\right]^{+}, \tag{9}$$

where $\boldsymbol{x}^{(t)}\in\arg\min_{\boldsymbol{x}\in X}\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda}^{(t)})$ and $\alpha>0$.

Now, observe that update (9) has a queue-like form, and that if we replace $\bar{\boldsymbol{b}}$ by a random variable $\boldsymbol{b}^{(t)}$, and $\boldsymbol{x}^{(t)}$ by $\boldsymbol{y}^{(t)}$ we can write

$$\boldsymbol{\mu}^{(t+1)}=\left[\boldsymbol{\mu}^{(t)}+\alpha\left(A\boldsymbol{y}^{(t)}+\boldsymbol{b}^{(t)}\right)\right]^{+}. \tag{10}$$

with $\boldsymbol{\lambda}^{(1)}=\boldsymbol{\mu}^{(1)}$. Further, if we divide $\boldsymbol{\mu}^{(t)}$ by α we have

$$\boldsymbol{Q}^{(t+1)}=\left[\boldsymbol{Q}^{(t)}+A\boldsymbol{y}^{(t)}+\boldsymbol{b}^{(t)}\right]^{+}, \tag{11}$$

yielding the queue updates as given in §3. This is equivalent to having dual subgradient updates *with perturbations*. In this case, these perturbations correspond to the noise introduced by using instantaneous data arrivals $\boldsymbol{b}^{(t)}$ instead of the real mean $\bar{\boldsymbol{b}}$. The convergence of this method has been established by Valls *et al.* in [38, Th. 1].

In this way, the connection between Problem 1 and Problem 2 comes from Lagrange duality: boundedness of the dual variables implies feasibility of the primal problem. To understand this, let us consider the deterministic setting for simplicity. In this case, the strong stability requirement ($\frac{1}{k}\sum_{t=1}^{k}\mathbb{E}(\boldsymbol{Q}^{(t)})\prec\infty$) corresponds to the queues being bounded for all k. From the dual subgradient we can write

3. Obtained as a result of solving a sequence of convex optimizations.

4. We refer the reader to §2 in [38] for a more detailed explanation.

5. This is always the case when the Slater condition is satisfied, i.e., there exists a point $\boldsymbol{x}\in X$ such that $A\boldsymbol{x}+\bar{\boldsymbol{b}}\prec 0$

$$\boldsymbol{\lambda}^{(k+1)} = \left[\boldsymbol{\lambda}^{(k)} + \alpha(A\boldsymbol{x}^{(k)} + \boldsymbol{b}^{(k)})\right]^{+} \succeq \boldsymbol{\lambda}^{(k)} + \alpha(A\boldsymbol{x}^{(k)} + \boldsymbol{b}^{(k)}).$$

Applying the latter recursion from $\boldsymbol{\lambda}^{k}$ to $\boldsymbol{\lambda}^{(1)}$ we have that

$$\boldsymbol{\lambda}^{(k+1)} \succeq \boldsymbol{\lambda}^{(1)} + \alpha \sum_{i=1}^{k} (A\boldsymbol{x}^{(i)} + \boldsymbol{b}^{(i)})$$

Next, divide by αk and use the fact that $\lambda^{(1)} = \mathbf{0}$ (for simplicity) to obtain

$$\frac{\boldsymbol{\lambda}^{(k+1)}}{\alpha k} \succeq \frac{1}{k}\sum_{i=1}^{k}(A\boldsymbol{x}^{(i)} + \boldsymbol{b}^{(i)}) = A\frac{1}{k}\sum_{i=1}^{k}\boldsymbol{x}^{(i)} + \frac{1}{k}\sum_{i=1}^{k}\boldsymbol{b}^{(i)}.$$

Now, as long as the difference $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\mu}^{(k)}\|_2 = \|\boldsymbol{\lambda}^{(k)} - \alpha\boldsymbol{Q}^{(k)}\|_2$ is uniformly bounded, we will have bounded $\boldsymbol{Q}^{(k+1)}$ provided $\boldsymbol{\lambda}^{(k+1)}/\alpha$ is also bounded. Therefore, when $k \to \infty$, we have that $A\bar{\boldsymbol{x}} + \bar{\boldsymbol{b}} \preceq 0$, being $\bar{\boldsymbol{b}} := \lim_{k\to\infty}\frac{1}{k}\sum_{i=1}^{k}\boldsymbol{b}^{(i)}$ and $\bar{\boldsymbol{x}} := \lim_{k\to\infty}\frac{1}{k}\sum_{i=1}^{k}\boldsymbol{x}^{(i)}$. Hence, we have the feasibility condition of Problem 2 as a result of the queues being bounded. For the sake of conciseness, we omit the details for the stochastic case (we need to show that $\mathbb{E}\left[\boldsymbol{\lambda}^{(k)}\right]$ increases at most at a rate $O(\sqrt{k})$) and instead refer the reader to [38, claim (iii)-(iv), Theorem 1]. Nonetheless, the rationale is the same as in the deterministic case.

As a result, the solution to Problem 2 provides a real-valued vector $\boldsymbol{x}$ indicating the fraction of time each link should be selected to solve Problem 1. Since $X \subseteq \mathrm{conv}(Y)$, we can always write $\boldsymbol{x}$ as a convex combination of points in Y and so devise an online algorithm that builds sequences satisfying $\boldsymbol{x}$. Let us illustrate this with an example. Imagine a BS with a single RAT, two users, and a single MCS available for both users. Assume that, at each slot, at least one packet arrives for each user. In this case, the solution to Problem 2 is obviously $\boldsymbol{x} = [0.5, 0.5]^T$ (i.e. equal share of time to both users). Then, $\langle \boldsymbol{y_1}, \boldsymbol{y_2}, \boldsymbol{y_1}, \boldsymbol{y_2}, \dots \rangle$ where $\boldsymbol{y_1} = [1, 0]^T$ and $\boldsymbol{y_2} = [0, 1]^T$ is evidently the corresponding solution to Problem 1 (e.g., if $T = 4$, $\boldsymbol{x} = \frac{1}{4}(\boldsymbol{y_1} + \boldsymbol{y_2} + \boldsymbol{y_1} + \boldsymbol{y_2})$). If, for instance, $\boldsymbol{y_1}$ is not admissible in the first two slots, then, the admissible sequence $\langle \boldsymbol{y_2}, \boldsymbol{y_2}, \boldsymbol{y_1}, \boldsymbol{y_1}, \dots \rangle$ will be a solution to Problem 1. While these discrete sequences can be found for this simple example in a trivial manner, it becomes a hard combinatorial problem to find them in more complex setups.

For all the above to hold, however, we need that (i) $\boldsymbol{b}^{(t)}$ is an i.i.d. stochastic process with finite variance and mean $\bar{\boldsymbol{b}}$ (i.e., $\lim_{t\to\infty}\frac{1}{t}\sum_{i=1}^{t}\boldsymbol{b}^{(i)} = \bar{\boldsymbol{b}}$), and ($ii$) the difference $\|\sum_{i=1}^{t}\boldsymbol{x}^{(i)} - \boldsymbol{y}^{(i)}\|_2$ is uniformly bounded. We will assume the former requirement is met and focus on designing an algorithm that satisfies the latter.

4.1.2 Design of the Objective Function

Our goal is to design a objective function f that balances infrastructure/operator cost and user satisfaction. To this aim, we map each flow i with a different utility $U_i(\rho_i)$ where ρ_i is the i'th element of $\boldsymbol{\rho} := A\boldsymbol{x} = [\rho_1, \dots, \rho_{2N}]^T$. We consider three types of utility. The first one is

$$U_i(\boldsymbol{x}) = g_i(\rho) := \frac{1 + e^{a_i b_i}}{e^{a_i b_i}}\left(\frac{1}{1+e^{-a_i(\rho - b_i)}} - \frac{1}{1+e^{a_i b_i}}\right), \tag{12}$$

a normalized sigmoidal-like function for delay-sensitive flows, where a_i and b_i are parameters. For example, when $a_i \approx b_i \approx \rho$ this function is a good approximation to a step function for voice traffic with rate requirement ρ; and when $a_i \ll \rho \ll b_i$ it can be used to model the utility of adaptive real-time applications with mean rate ρ [42]. The second utility function we consider is

$$U_i(\boldsymbol{x}) = h_i(\rho) := \frac{\log(1 + c_i \cdot \rho)}{\log(1 + c_i \cdot \hat{\rho})}, \tag{13}$$

which is useful for elastic (delay-tolerant) flows. Parameter $\hat{\rho}_i$ is the maximum aggregated throughput achievable in the system, and c_i is the *satisfaction* growth rate per ρ allocated. Finally, we also consider

$$U_i(\boldsymbol{x}) = 1, \tag{14}$$

which captures the case where flows do not require QoS guarantees.

In addition, we want to give operators flexibility in the way their infrastructure is utilized. For instance, an operator may want to aggregate system load into the minimum possible subset of RATs in order to save costs. This can be done by assigning a weight $w_{r,m}$ on each PRB allocated in RAT r when modulated with index m. Although, we advocate for a simple linear cost function to accommodate operator preferences, any convex function can be supported. The resulting objective function is the following:

$$f(\boldsymbol{x}) = \eta \boldsymbol{w}^T \boldsymbol{x} - \frac{1}{N}\sum_{i=1}^{N}\log(U_i(\boldsymbol{x})) \tag{15}$$

where $\eta \geq 0$ is a parameter that controls the relative importance of system cost reduction against overall utility satisfaction. That is, for a given η, a solution $\boldsymbol{x}^*$ to Problem 2 corresponds to a point in the Pareto optimal trade-off between utility satisfaction and cost minimization. For example, by setting $\eta = 0$ we would only consider QoS satisfaction irrespective of how the infrastructure is used.

The convexity of (15) is proved in the following lemma.

Lemma 1. *Function $f(\boldsymbol{x})$ in Eq. (15) is convex for $\boldsymbol{x} \succeq \mathbf{0}$.*

Proof: We proceed by showing that f is the sum of two convex functions. The first term, $\eta\boldsymbol{w}^T\boldsymbol{x}$, is linear in $\boldsymbol{x}$ and so convex. We study three cases for the second term:

(i) ($U_i(\boldsymbol{x})$ is equal to (12)). Let $c := (1 + e^{a_i b_i})(e^{a_i b_i})^{-1}$, $d := e^{a_i b_i}$ and note that they are both strictly positive for any $a_i, b_i \in \mathbf{R}$. With these change of variables and after some manipulations we can write

$$-\log(g_i(\rho)) := -\log(c) - \log\left(\frac{e^{a_i\rho}}{e^{a_i\rho_i} + d} - \frac{1}{1+d}\right).$$

The first term on the RHS of the last equation is a constant, and the second term is the composition of a convex non-increasing function (i.e. $-\log$) with a non-negative concave function, and so convex [41].

(ii) ($U_i(\boldsymbol{x})$ is equal to (13)). This case is immediate since $-\log(g_i(\rho))$ is, again, the composition of a convex non-increasing function with a concave function [41].

(iii) ($U_i(\boldsymbol{x}) = 1$). The function does not depend on $\boldsymbol{x}$ and so it is constant. □

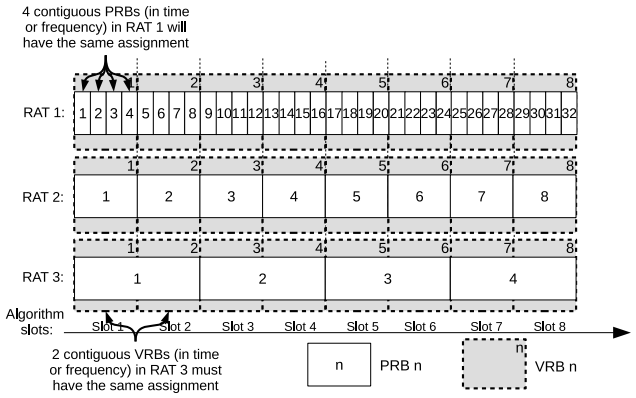


Fig. 6. Three RATs with capacity 32, 8, and 4 PRBs, respectively, per time×bandwidth, like the example shown in Fig. 3a. One VRB in RAT 1 contains four PRBs with equal allocation. One PRB in RAT 3 contains two VRBs which must have equal allocation.

4.2 LaSR Scheduling Algorithm

We now present the main contribution of our paper: an online algorithm that selects optimal discrete actions based on the previous model. We remark that, while some prior work on scheduling problems relies on non-trivial approximations of the convex formulation presented earlier, our algorithm strictly maps fluid actions into optimal discrete actions preserving stability guarantees.

The goal of our algorithm is to solve Problem 1, i.e., (i) to make discrete actions $\boldsymbol{y}^{(t)} \in \pi$ that preserve convergence and stability, and (ii) to include restrictions on how actions are taken. The latter is required to give us enough flexibility to make *admissible* choices that accommodate real-system constraints including (but not limited to):

- (i) Protocols to signal users their resource assignments may trade off granularity with reduced overhead, imposing constraints on how actions can be taken, e.g., PRBs may be grouped—see Fig. 2 in §2;
- (ii) If we decide to turn off a RAT, we need to guarantee that subsequent actions do not schedule resources on the deactivated RAT for at least the amount of time required to turn it back on again. The same applies when a specific modulation is not available due to e.g., fading, or when a RAT is unavailable due to an off cycle when operating in unlicensed spectrum;
- (iii) Finally, the heterogeneity across RATs in the mapping between VRBs and PRBs may also require scheduling restrictions.[6] This is illustrated in Fig. 6 with an example depicting 3 RATs with capacity 16, 8, and 4 PRBs, respectively, per time×bandwidth unit. In this case, one VRB maps exactly one PRB in RAT 2. This implies that one VRB in RAT 1 contains 2 PRBs with equal allocation (with some loss in granularity in this case), and that one PRB in RAT 3 contains 2 VRBs, which shall have equal allocation, i.e., a sequence $\pi = \langle \dots, \boldsymbol{y}^{(t)}, \boldsymbol{y}^{(t+1)}, \dots \rangle$ that makes a different schedule choice in two consecutive VRBs t and $t+1$ in RAT 3 (containing one PRB) would not be admissible, $\pi \notin \Pi$.

Classic stochastic control approaches [35]–[37] do not provide enough flexibility to adapt to this type of real-system constraints in a trivial manner. We therefore resort to a new technique based on low-complex convex optimization that lets us decouple the choice of subgradient from the selection of a (discrete) control actions [38]. This is important because we can design scheduling policies that provably converge to an optimal point without having to specify the nature of the underlying system constraints. In this way, we can design a single *supple* algorithm that jointly handles all aforementioned practical limitations in a general yet effective manner.

6. This mapping is a design choice that depends on the computational capacity available, e.g., a VRB may contain several PRBs spanning longer time slots to alleviate computational burden. Importantly, the design of our algorithm adapts to this choice with no loss in optimality.

Algorithm 1 LaSR algorithm

1: $\boldsymbol{W} := \left[\boldsymbol{y}_1, \dots, \boldsymbol{y}_{|Y|}\right]$ ▷ Collection of points in Y
2: $E := \left\{\boldsymbol{e}_1, \dots, \boldsymbol{e}_{|Y|}\right\}$ ▷ $|Y|$-dimensional standard basis
3: $U := \mathrm{conv}(E)$ ▷ Convex hull of E
4: $\boldsymbol{\delta}^{(1)} \leftarrow \boldsymbol{0}$
5: $\boldsymbol{\lambda}^{(1)} \leftarrow \boldsymbol{0}$
6: **for each slot** $t = 1, 2, \dots$ **do**
7: SET-STATE$\left(t, \boldsymbol{\delta}^{(t)}\right)$ ▷ Activate/deactivate RATs
8: $\boldsymbol{x}^{(t)} \leftarrow \text{SOLVER}\left(\boldsymbol{\lambda}^{(t)}\right)$ ▷ Solution to Problem 3
9: $\boldsymbol{u}^{(t)} \in \arg\min_{\boldsymbol{u} \in U} \|\boldsymbol{x}^{(t)} - \boldsymbol{W}\boldsymbol{u}\|_2^2$
10: $\hat{E}^{(t)} \subseteq E$ ▷ Current available set of actions
11: $\boldsymbol{e}^{(t)} \in \arg\min_{\boldsymbol{e} \in \hat{E}^{(t)}} \|\boldsymbol{\delta}^{(t)} + \boldsymbol{u}^{(t)} - \boldsymbol{e}\|_\infty$
12: $\boldsymbol{\delta}^{(t+1)} \leftarrow \boldsymbol{\delta}^{(t)} + \boldsymbol{u}^{(t)} - \boldsymbol{e}^{(t)}$
13: $\boldsymbol{\lambda}^{(t+1)} = \left[\boldsymbol{\lambda}^{(t)} + \alpha\left(A\boldsymbol{x}^{(t)} + \boldsymbol{b}^{(t)}\right)\right]^+$
14: $\boldsymbol{y}^{(t)} \leftarrow \boldsymbol{W}\boldsymbol{e}^{(t)}$ ▷ Discrete solution
15: **end for**

To start, let us define $\boldsymbol{W} := [\boldsymbol{y}_1, \dots, \boldsymbol{y}_{|Y|}]$, $E := \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_{|Y|}\}$, where $\boldsymbol{W}$ is a collection of all points in Y (as column vectors), and e_i is an $|Y|$-dimensional standard basis vector, i.e., all elements of vector e_i are equal to 0 except element i which is equal to 1. Since we can write any point $\boldsymbol{x} \in X$ as a convex combination of points in Y, i.e., $\boldsymbol{x} = \boldsymbol{W}\boldsymbol{u}$, $\boldsymbol{u} \in U$, there always exists a vector $\boldsymbol{e} \in E$ such that $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{e}$. Vector $\boldsymbol{u}$ can be obtained by computing $\arg\min_{\boldsymbol{u} \in U} \|\boldsymbol{x} - \boldsymbol{W}\boldsymbol{u}\|_2^2$. Then, by [38, Th. 2], for any sequence $\langle \boldsymbol{u}^{(t)} \rangle$ of points from U, there exists a sequence $\langle \boldsymbol{e}^{(t)} \rangle$ of points from E such that $\| \sum_{i=1}^{t} \boldsymbol{u}^{(i)} - \boldsymbol{e}^{(i)} \|_2$ is uniformly bounded. With this in mind, our job is to find a method that constructs sequences of discrete actions $\langle \boldsymbol{e}^{(t)} \rangle$ that satisfies the aforementioned condition while being admissible at the same time.

We present our approach in Algorithm 1. We use two auxiliary vectors: $\boldsymbol{\lambda}^{(t)}$ and $\boldsymbol{\delta}^{(t)} \in \mathbb{R}^{|Y|}$. $\boldsymbol{\lambda}^{(t)}$ is used in the subgradient method solving Problem 3. $\boldsymbol{\delta}^{(t)}$ maintains the aggregate deficit or surplus of fluid actions taken when mapping $\boldsymbol{x}^{(t)}$ to $\boldsymbol{y}^{(t)}$ and it is the key to accommodate practical system constraints and the mapping of VRBs into actual physical resources with no loss in optimality. In step (7), function SET-STATE() (de)activates one or more RATs if needed. Different policies can be applied. For instance, if a RAT is not allocated data between slot t and slot $t+T$ (i.e., $\boldsymbol{\delta}$ is equal to zero on links related to a RAT during this window of slots) then we can deactivate it. Conversely, if one or more links related to a RAT in $\boldsymbol{\delta}^{(t)}$ at any slot t become nonzero, this RAT needs to be restarted. This policy is shown in Algorithm 2. In step (8) we use a standard solver for Problem 3 with $\boldsymbol{\lambda} = \boldsymbol{\lambda}^{(t)}$ to obtain the optimal (fluid) schedule $\boldsymbol{x}^{(t)}$ at this time. We then map $\boldsymbol{x}^{(t)}$ into a feasible $\boldsymbol{u}^{(t)}$ such that $\boldsymbol{x}^{(t)} = \boldsymbol{W}\boldsymbol{u}^{(t)}$ in step (9). This can be computed by solving a linear system of equations (see [41,

Algorithm 2 Activation/Deactivation algorithm

1: **procedure** SET-STATE(t, δ)
2: $T \in \mathbb{Z}_+$ ▷ Epoch interval to make sleep decisions
3: $C \in \{0,1\}^{R \times L}$ ▷ C maps links to RATs
4: $\boldsymbol{m} \in \{0,1\}^L$, $\hat{\boldsymbol{m}} \in \{0,1\}^L$
5: $\boldsymbol{m} \leftarrow \mathbf{0}^L$
6: **for** $i \in \{1, \ldots, |\boldsymbol{\delta}|\}$ **do**
7: $\boldsymbol{\epsilon} \leftarrow \mathbf{0}^{|\boldsymbol{\delta}|}$
8: $\epsilon_i \leftarrow \delta_i$
/ Function $\mathbb{1}(\cdot)$ returns 1 if $(\cdot)$ is true and 0 otherwise. $\boldsymbol{m}$ is an L-dimensional vector where the i'th element is 1 if some resources are debited to link i. */*
9: $\boldsymbol{m} \leftarrow \boldsymbol{m} \vee \mathbb{1}(W\boldsymbol{\epsilon} \succ 0)$ ▷ $\vee$ is the OR operator
10: **end for**
/ $\hat{\boldsymbol{m}}$ is an L-dimensional vector where the i'th element is 1 if link i has not been used within the current T-interval */*
11: $\hat{\boldsymbol{m}} \leftarrow \hat{\boldsymbol{m}} \wedge (\neg \boldsymbol{m})$ ▷ $\wedge$/$\neg$ are AND/NOT operators
/ RATs (in sleep state) required for allocation are awaken. */*
12: SET-AWAKE($\mathbb{1}(C\boldsymbol{m} \succ 0)$)
/ Every epoch, all RATs that have not been required within the T-interval are sent to sleep. */*
13: **if** $t \pmod T = 0$ **then**
14: SET-SLEEP($\mathbb{1}(C\hat{\boldsymbol{m}} \succ 0)$)
15: $\hat{\boldsymbol{m}} \leftarrow \mathbf{1}^L$
16: **end if**
17: **end procedure**

§6.2]). Now, in step (10), we let $\hat{E}^{(t)} := \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{|\boldsymbol{\mathcal{A}}^{(t)}|}\}$ be a set containing all basis vectors such that an admissible action at time t $\boldsymbol{y} \in \boldsymbol{\mathcal{A}}^{(t)}$ is equal to $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{e}$ with $\boldsymbol{e} \in \hat{E}^{(t)}$. As we explained above, the set of available actions $\boldsymbol{\mathcal{A}}^{(t)}$ depends on the RAT and the nature of action unavailability (e.g., a CQI report indicating a modulation is not currently available, an unlicensed MAC controller alerting the current slot corresponds to an off period, a hardware controller notifying a RAT is not yet activated, or our DCI mapper indicating of some signaling constraints). Note also that, based on the previous step, $\hat{E}^{(t)}$ will not contain any action i that involves the deactivated RAT while the RAT is off or during the time it takes for the RAT to be fully operational after the activation process is started. We also use $\hat{E}^{(t)}$ to support RATs with heterogeneous capacities in terms of available PRBs per time×bandwidth unit. For instance, in the example of Fig. 6 (and Fig. 3a), we should not include in $\hat{E}^{(t)}$, in slots $\{2, 4, 6, 8\}$, any action that implies an allocation on RAT 3 different than the one in slots $\{1, 3, 5, 7\}$, respectively. This allows us to apply our homogeneous model based on VRBs onto a heterogeneous multi-RAT system based on PRBs, with no loss in optimality. Note that if all links (all modulation levels and all RATs) are available at this time, $\hat{\boldsymbol{\delta}}^{(t)} = \boldsymbol{\delta}^{(t)}$ and $\hat{\boldsymbol{u}}^{(t)} = \boldsymbol{u}^{(t)}$. Then, in step (11), we select an action out of the currently admissible set of actions $\hat{E}^{(t)}$ by computing $\boldsymbol{e}^{(t)} \in \arg\min_{\boldsymbol{e} \in \hat{E}^{(t)}} \|\boldsymbol{\delta}^{(t)} + \boldsymbol{u}^{(t)} - \boldsymbol{e}\|_\infty$ Finally, steps (12) and (13) maintain the surplus/deficit of fluid resources by computing $\boldsymbol{\delta}^{(t+1)} = \sum_{i=1}^t \boldsymbol{u}^{(i)} - \boldsymbol{e}^{(i)}$, and update the Lagrange multipliers based on current queue states $\boldsymbol{b}^{(t)}$, respectively. Finally, the current action is selected in step (14) by computing $\boldsymbol{y}^{(t)} = \boldsymbol{W}\boldsymbol{e}^{(t)}$.

4.3 Performance Analysis

As explained above, it is important that Algorithm 1 preserves that $\| \sum_{i=1}^t \boldsymbol{u}^{(t)} - \boldsymbol{e}^{(t)}\|_2$ is uniformly bounded. We prove this in the next lemma, which is an extension of [38, Theorem 5] to consider a time-varying action set.

Lemma 2. *Let $\hat{E}^{(t)} \subseteq E$ be a set that maps the admissible control actions at time slot t. Suppose the number of consecutive slots during which an action is not available in $\hat{E}$ is upper bounded by $\sigma \geq 0$. Then Algorithm 1 keeps difference $\| \sum_{i=1}^t \boldsymbol{u}^{(i)} - \boldsymbol{e}^{(i)}\|_2$ uniformly bounded for all t.*

Proof: We take as a starting point the result of Theorem 5 in [38]. This lemma says that for two arbitrary sequences of actions from U and E we always have $\|\boldsymbol{\delta}^{(t)}\|_2 \leq \gamma^{(t)}\sqrt{|Y|}(|Y| - 1)$, where $\gamma^{(t)} := -\min_{\kappa \in \{1,\ldots,|Y|\}} \delta_\kappa^{(t)}$. We want to show that by selecting actions from the constrained set $\hat{E}^{(t)}$, we can construct a sequence that keeps $\gamma^{(t)}$ bounded.

We first show that a set $\hat{E}^{(t)}$ satisfying the conditions of the lemma always exists. Observe that for any vectors $\boldsymbol{u}^{(i)} \in U$ and $\boldsymbol{e}^{(i)} \in E$ we have that $\mathbf{1}^T\boldsymbol{u}^{(t)} = \mathbf{1}^T\boldsymbol{e}^{(t)} = 1$ with all elements of $\boldsymbol{u}^{(t)}$ and $\boldsymbol{e}^{(t)}$ being nonnegative. Also, since $\boldsymbol{\delta}^{(t)} = \sum_{i=1}^t \boldsymbol{u}^{(i)} - \boldsymbol{e}^{(i)}$, then $\mathbf{1}^T\boldsymbol{\delta}^{(t)} = 0$, and therefore $\boldsymbol{\delta}^{(t)}$ is either (i) a vector where all components are zero, or (ii) has at least one element that is strictly positive (and one strictly negative). That is, $\delta_\kappa^{(t)} \geq 0 \geq -\sigma$ for at least one $\kappa \in \{1, \ldots, |Y|\}$, and so it is sufficient for set $\hat{E}^{(t)}$ to contain an action $\boldsymbol{e}_\kappa$ to satisfy the condition of the lemma.

We are now in position to show that the greedy update $\boldsymbol{e}^{(t)} \in \arg\min_{\boldsymbol{e} \in \hat{E}^{(t)}} \|\boldsymbol{\delta}^{(t)} + \boldsymbol{u}^{(t)} - \boldsymbol{e}\|_\infty$ keeps $\gamma^{(t)}$ uniformly bounded. First, observe that if $\hat{E}^{(t)} = E$ then $\boldsymbol{e}^{(t)}$ is chosen to decrease the largest component of vector $(\boldsymbol{\delta}^{(t)} + \boldsymbol{u}^{(t)})$ by 1, and when $\hat{E}^{(t)}$ is a subset of E we will have the same behavior but with the restriction of the actions available. Hence, if $\delta_j^{(\tau)} \geq -\gamma^{(\tau)} \geq -(\sigma + 1)$ at some $\tau \in \mathbb{N}$ for all $j \in \{1, \ldots, |Y|\}$, our update will select an action from $\hat{E}^{(\tau)}$ such that $\delta_j^{(\tau+1)} \geq -\gamma^{(\tau)} \geq -(\sigma+1)$. That is, $\gamma^{(t)} \leq \sigma+1$ for all $t \geq \tau$. To conclude, observe that since $\delta_j^{(t)} = 0 \geq -(\sigma+1)$ when $t = 1$ we will have that $\gamma^{(t)}$ is bounded for all $t \geq 1$. □

The above lemma proves that Algorithm 1 guarantees that $\| \sum_{i=1}^t \boldsymbol{u}^{(t)} - \boldsymbol{e}^{(t)}\|_2$ is uniformly bounded while the following theorem proves the optimality of our algorithm to solve Problem 1.

Theorem 1 (Optimality of Algorithm 1)**.** *Algorithm 1 preserves $\left\| \sum_{i=1}^t A\boldsymbol{x}^{(i)} - A\boldsymbol{y}^{(i)} \right\|_2$ uniformly bounded.*

Proof: Recall that $\boldsymbol{x}^{(t)} = \boldsymbol{W}\boldsymbol{u}^{(t)}$ and $\boldsymbol{y}^{(t)} = \boldsymbol{W}\boldsymbol{e}^{(t)}$; then the absolute error of data units allocated to each user is bounded by $\left\| \sum_{i=1}^t A\boldsymbol{x}^{(i)} - A\boldsymbol{y}^{(i)} \right\|_2 \leq A\|\boldsymbol{W}\|_2 \left\| \sum_{i=1}^t \boldsymbol{u}^{(i)} - \boldsymbol{e}^{(i)} \right\|_2$ Then, it is easy to see that, from Lemma 2, Algorithm 1 guarantees that

$$\left\| \sum_{i=1}^t A\boldsymbol{x}^{(i)} - A\boldsymbol{y}^{(i)} \right\|_2 \leq A\gamma^{(t)}\|\boldsymbol{W}\|_2\sqrt{|Y|}(|Y| - 1) \quad (16)$$

is uniformly bounded. □

The above theorem proves that LaSR guarantees a scheduling policy $\pi = \langle \boldsymbol{y}^{(t)} \rangle$ within a ball around the optimal fluid solution. In this way, Algorithm 1 solves Problem 1 by processing a series of polynomial complex tasks in each slot, dominated by the complexity of the convex solver of choice to execute step (8) and (9) ($\mathcal{O}(\sqrt{L}\ln(1/\epsilon))$ [43] in case of an interior-point method) plus that of the sorting algorithm used to execute step (11) ($\mathcal{O}(L \log L)$).

5 Performance Evaluation

In this section we illustrate the most important features of LaSR via simulations in a plurality of scenarios. As a benchmark, we will use in some of our experiments an ideal scheduler that is allowed to make (optimal) unconstrained fluid scheduling actions, obtained by directly applying the solution to Problem 3, and the Queue Side Greedy (QSG) PRB scheduling algorithm proposed in [21] which, to the best of our knowledge, is the only work with carrier aggregation support comparable, in its applicability to real systems, to ours (see §2 for related literature and how it differs to our work). The approach of [21] is however a heuristic with no performance guarantees. QSG is designed to minimize head-of-line (HOL) delay, so that it allocates resources first to those users with longer waiting time.

For simplicity, and unless otherwise stated, we carry out most of our tests in a scenario with 3 heterogeneous RATs with fixed bandwidth of 10 MHz (e.g. an LTE macro-cell), 5 MHz (e.g. an LTE small-cell) and 20 MHz (e.g. a mmWave cell), respectively. Without loss of generality, we assume LTE numerology for all RATs: PRBs are 180KHz/slot—50, 25, and 100 PRBs/slot, respectively (some subcarriers are used for control) and one transmission time interval (TTI) is 1 ms. Each VRB contains $1, 0.5$ and 2 PRBs, respectively. We also impose 8 and 1280 ms delay to activate and deactivate, respectively, a non-primary RAT (see [44]).[7] Finally we use the next cost vector $\boldsymbol{w}$ in most of our simulations:

$$\begin{matrix}\text{Macro}\\ \text{Small}\\ \text{mmWave}\end{matrix}\begin{pmatrix}1\\2\\6\end{pmatrix}\text{bit}^{-1}. \tag{17}$$

The rationale behind this choice is to illustrate a hierarchical preference when activating RATs: we favor the assignment of resources from the "Macro" first (the primary RAT), "Small" second, and finally "mmWave". These costs are selected for illustration purposes only; they can be tuned depending on the actual deployment.

Our goal in the sequel is to assess the ability of LaSR to:

1) Stabilize queues (maximize throughput) regardless of arrival user traffic patterns (§5.1);
2) Find an optimal solution to Problem 1 in heterogeneous multi-RAT systems (§5.1, §5.2) with QoS guarantees (§5.3, §5.4) and practical constraints (§5.8);
3) Balance system/operator costs (§5.1, §5.2) and user preferences (§5.3, §5.4);
4) Converge quickly upon changes on the network (users joining/leaving the system in §5.2, §5.3 user mobility in §5.5), and upon imperfect information (§5.6);
5) Accommodate practical considerations such as variable and uncertain bandwidth (§5.7) and constraints on the way scheduling actions shall be taken (§5.8, §5.9).

5.1 Heterogeneous static scenario

We start with the simple 3-RAT heterogeneous scenario introduced above with 5 traffic classes where users are located such that the average channel conditions between users and RATs are identical for all users. In addition, in this example we do not enforce QoS criteria across flows, that is, $U_i(\cdot) = 0$ for all flows. For clarity of illustration, we allow a reduced set of 3 MCS indexes for each RAT:

$$\begin{matrix}\\ \text{Macro}\\ \text{Small}\\ \text{mmWave}\end{matrix}\begin{matrix}\begin{matrix}m_0 & m_1 & m_3\end{matrix}\\ \begin{pmatrix}0.2344 & 2.1602 & 5.5547\\ 0.2344 & 2.1602 & 5.5547\\ 0.5 & 2.0 & 4.875\end{pmatrix}\end{matrix}\text{bps/Hz}.$$

This limited set allows us to build very simple scenarios which will help us to better understand the features of our algorithm. The modulations for the macro and the small-cell are a subset from LTE's modulation and coding scheme (MCS) indexes $\{0, 14, 28\}$ [45], respectively, and the modulations used for the mmWave RAT, $\{0, 9, 23\}$, are obtained from a subset proposed by EU project MiWEBA [46].

The goal of this test is twofold. First, to validate that LaSR stabilizes all queues up to reaching the boundary of the system capacity[8] regardless of the traffic pattern (Fig. 8), i.e., it maximizes throughput. Second, to show that LaSR performs optimally, according to our cost function,

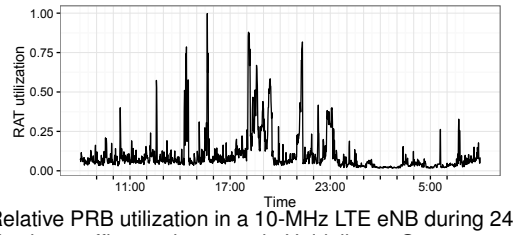
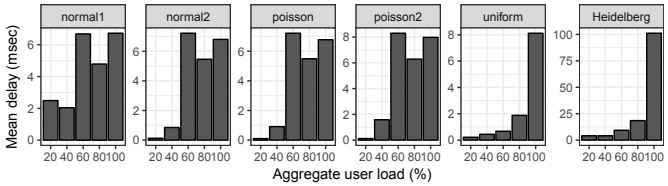
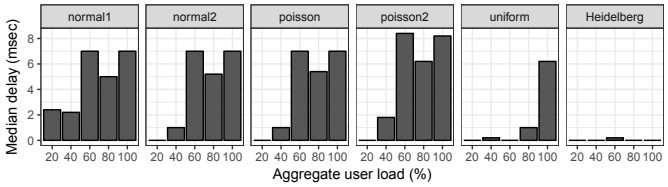


Fig. 7. Relative PRB utilization in a 10-MHz LTE eNB during 24 hours in a weekday in an office environment in Heidelberg, Germany.



(a) Mean user delay.



(b) Median user delay.

Fig. 8. Heterogeneous scenario with different traffic distributions. Delay experienced by each user.

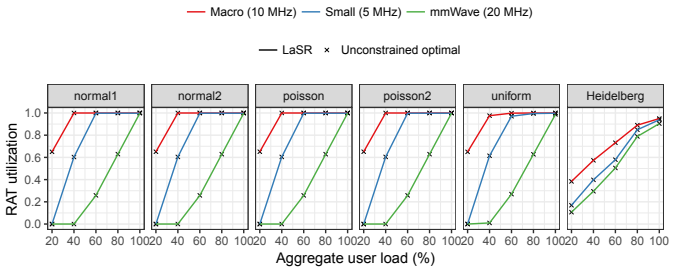
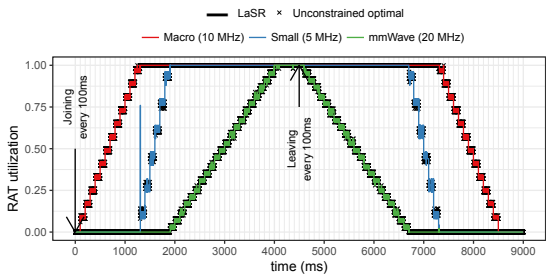


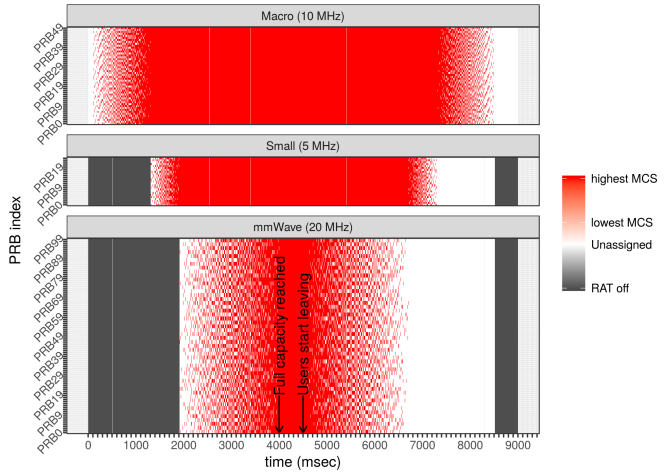
Fig. 9. Heterogeneous scenario with different traffic distributions. Relative usage of PRB resources per RAT.

7. RAT (de)activation involves only its PHY layer as the MAC and remaining upper layers are common across the multiple RATs of the BS, i.e., we do not consider (de)activation of the whole BS.

8. We define capacity as the maximum aggregate bitrate achievable in the system, i.e. using all RATs at highest modulation schemes.

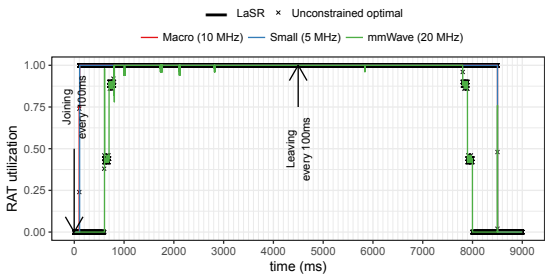


(a) Relative usage of PRB resources of each RAT over time.

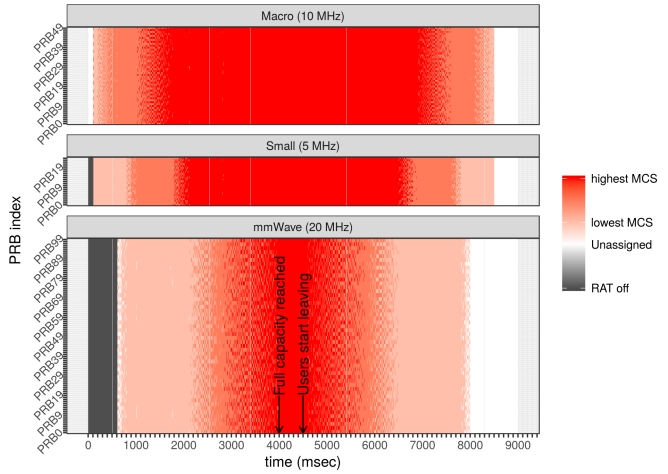


(b) State and MCS of each PRB and time slot for all RATs.

Fig. 10. Users join and leave every 100 ms. Costs defined in Eq. (17)



(a) Relative usage of PRB resources of each RAT over time.



(b) State and MCS of each PRB and time slot for all RATs.

Fig. 11. Users join/leave every 100 ms. Costs defined in Eq. (18)

in all load regimes (Fig. 9). As an example, we use four synthetic patterns with mean load $\bar{b}_i$ to model the arrival of user data, namely "normal1" for a normal distribution $\boldsymbol{b}^{(t)} \sim \mathcal{N}(\bar{b}_i, 10)$ bits/TTI, "normal2" for a normal distribution $\boldsymbol{b}^{(t)} \sim \mathcal{N}(\bar{b}_i, 100)$ bits/TTI, "poisson" for a Poisson distribution $\boldsymbol{b}^{(t)} \sim \mathcal{P}(\bar{b}_i)$ bits/TTI, and "uniform" for an uniform distribution $\boldsymbol{b}^{(t)} \sim \mathcal{U}(0, 2\bar{b}_i)$ bits/TTI. For completeness, we have also included a scenario where each user gets a different share, equal to $\{10, 15, 20, 25, 30\}$ percent of the aggregate load each, following a Poisson distribution, and labelled as "poisson2". In addition, we collected 24-hour traces of traffic from a 10-MHz LTE eNB near NEC premises in Heidelberg, Germany, during a weekday. The relative PRB utilization of the eNB over time is depicted in Fig. 7. To this means, we have used `owl`, an LTE control channel decoder built on top of `srsLTE`, an open source LTE library for software defined radio [47], [48]. For this experiment, we have selected a 10-min sample with mean load equal to 13.14% relative to the capacity of our 3-RAT scenario and use this data as a fifth traffic pattern, labeled as "Heidelberg". In order to use this data for different loads, we have scaled up/down the individual loads with a proportional factor.

Fig. 8 depicts the mean and median user delay for different aggregate mean loads (relative to the system capacity) and traffic patterns. We can observe that the average delays are bounded for all load regimes, regardless the traffic pattern, validating our stability guarantees. Note moreover that the delay performance is very similar across distributions except for the traces we have collected ("Heidelberg"). The reason is that, as depicted in Fig. 7, the arrival process is very bursty, with very low load most of time and very short instances with very high load, leading to an increase in mean delay, while its median is contained. It is also worth remarking that we have not specified any QoS criteria across users in this experiment, which we evaluate later on.

Furthermore, Fig. 9 depicts the mean relative utilization of each RAT. In this case, we also compare the performance of LaSR with an ideal scheduler that is allowed to make (optimal) unconstrained fluid scheduling actions (labelled "Unconstrained optimal"). The latter has been obtained by directly applying the solution to Problem 3 (which gives us a benchmark to compare to). We observe that, according to our costs, the macro-cell is preferred until the capacity of this RAT is reached; then, the small-cell is activated. Note that the capacity of the small-cell is quickly reached and then the mmWave RAT is activated. This behavior is however different for "Heidelberg" trace. The reason is that the short bursts of very high load cause the system to fully use all RATs during the peak and leave them practically unused during the rest of the time. Remarkably, the results for LaSR follow very closely those of the ideal "Unconstrained optimal" scheduler, validating the optimality of our scheduler. We do not show results for QSG in this experiment because QSG does not consider RAT deactivation nor does it support setting preferences on the utilization of RATs.

5.2 Users joining and leaving

Hereafter, for simplicity, we will use the "poisson" model. The next experiment evaluates how quickly the controller adapts when users join and leave the system and illustrates the effect of different operational costs. We deploy the scenario with the 3 heterogeneous RATs we used earlier; in this case, however, the users (with the same homogeneous

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2018.2876847, IEEE Transactions on Mobile Computing

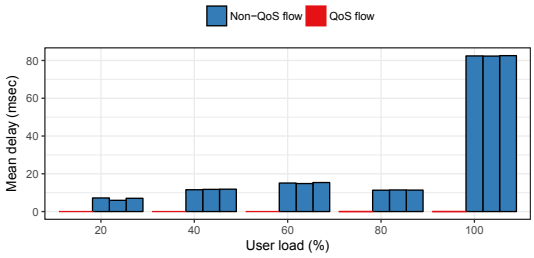


Fig. 12. Static setup with a mix of elastic and inelastic flows. Mean delay for each user type (inelastic vs. elastic) for different loads

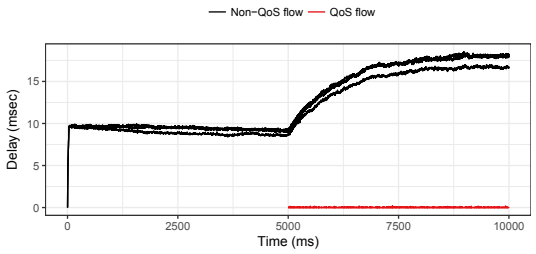


Fig. 13. Dynamic setup with 3 non-QoS flow types until $t = 5000$ where 2 inelastic flow types join the system. Figure shows delay experienced by each user as a function of time.

channel characteristics as before) join and leave at different times. From $t = 0$ without any user present, new users join the system every 100 ms (very fast) leading to a load increment of 2.5% of the total capacity. Then, a short time after the capacity boundary is reached, users start leaving at the same pace they arrived.

Fig. 10a depicts the mean RAT utilization, i.e., relative number of PRBs allocated in each RAT, as time evolves (and users join and leave) for LaSR (with colored lines) and the ideal benchmark "Unconstrained optimal" used before. We first note that there is a perfect match between LaSR and the benchmark and the system timely adapts to load changes in the system as users join and leave. Secondly, we observe that, according to the costs defined in Eq. (17), LaSR first uses resources from "Macro" until it is fully loaded. Then it begins assigning resources from "Small" which quickly saturates due to the low bandwidth of this RAT and then the scheduler starts using "mmWave". This demonstrates the flexibility of our approach to set system preferences.

Fig. 10b presents more detailed information of the same experiment, namely, a grid with the state of each PRB and RAT (y axis) as time goes on (x axis). Each element of the grid (each PRB) is colored depending on its assignment: black if the RAT is off, white if it is not assigned, and a gradient between pale red if the PRB is modulated with the lowest level available and bright red if the highest MCS is used. Similarly to Fig. 10a, Fig. 10b illustrates how both "Small" and "mmWave" are fully off until they are really needed. It is worth remarking that in low to mid load regimes, PRBs alternate between no assignment and highest MCS (no gradient). This is because the costs defined in Eq. (17) do not discriminate across MCS levels.

In contrast, Fig. 11 shows results with costs:

$$\begin{array}{c} \\ \text{Macro} \\ \text{Small} \\ \text{mmWave} \end{array} \begin{array}{c} \begin{array}{ccc} m_0 & m_1 & m_2 \end{array} \\ \begin{pmatrix} 1 & 5 & 10 \\ 2 & 10 & 20 \\ 6 & 30 & 60 \end{pmatrix} \end{array} \text{bit}^{-1} \tag{18}$$

Fig. 11a does not show a gradual increase of RAT utilization but rather all PRBs are used as soon as the RAT is needed. However, differently to the previous case, as depicted in Fig. 11b, although all PRBs are used—even with low- mid-load ranges—the MCSs (and therefore the transmission power) used on these PRBs have low to medium levels. In this way the traffic load is better distributed across all the bandwidth and therefore the same amount of load is processed with lower inter-cell interference as compared to the case of Fig. 10. This illustrates the ability of our controller to adapt to the preferences of each operator.

5.3 QoS with inelastic flows

We now evaluate heterogeneous QoS requirements in both static and dynamic environments. First, in Fig. 12 we illustrate the delay of 5 different flows in the static setup of §5.1. The difference now is that 2 of those flows have a sigmoidal-like utility function with parameters $a = 1$ kbps and b equal to their mean individual load. In addition, we set $\eta = 10^{-3}$ (see Eq. (15)). The results show how LaSR trades delay off from non-QoS flows to favor QoS flows. Secondly, we present in Fig. 13 a scenario where 3 non-QoS flow types share the system until $t = 5000$ when 2 new QoS flow types (with the same utility model as in the previous case) join the system. Each flow demands 19% bitrate of the overall capacity. The goal of this experiment is to show how quickly the system adapts to the new environment with two different traffic classes. We do not present results with QSG because it does not support QoS differentiation.

5.4 QoS with elastic flows

In the next experiment we set up a scenario with 2 co-located RATs with capacity 20 MHz each and 20 users divided into two groups. The first group, with 10 users, have poor average channel conditions, so they can only use modulation indexes $\{11, 12, 13\}$, whereas the 10 users in the second group have good average conditions and so can use modulation indexes $\{21, 22, 23\}$ [45]. The mean rate of each flow is 1 Mbps. Our goal is to evaluate short- and long-term fairness with elastic flows. To this aim we measure the individual throughput achieved by each user (with equal long-term mean data rate requirements) in different time windows, and calculate Jain's fairness index.[9] Computing a fairness index across short time windows helps us to understand short-term fairness performance [50]. In Fig. 14

9. Jain's fairness index is a well known metric to evaluate how *equitable* a resource allocation is, spanning from 1 (equal share) to $1/n$ where n is the number of users [49].

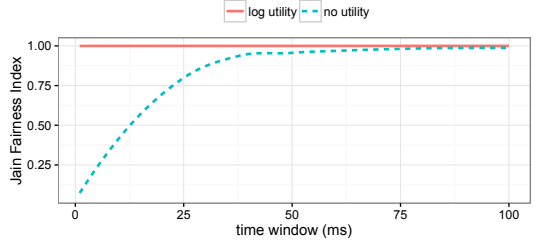
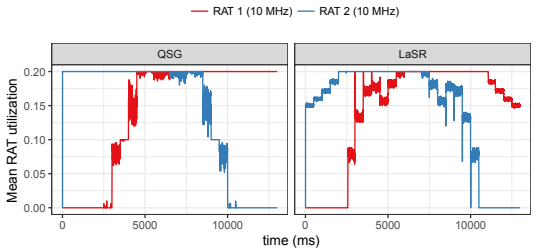
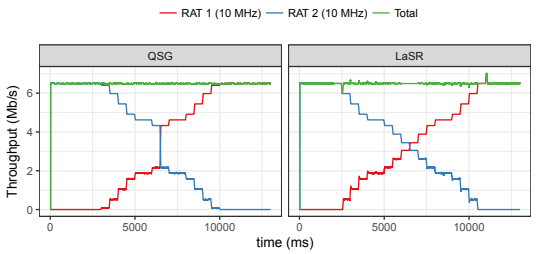


Fig. 14. Jain's Fairness Index [49] for user rates measured along different time windows. Short(long) time windows represent a measure of short(long)-term fairness.



(a) RAT utilization



(b) Throughput

Fig. 15. A set of 5 traffic flows jointly moving from RAT 1 to RAT 2. Figure shows average number of PRBs relative to the bandwidth of each RAT over time (top) and mean throughput per RAT and aggregate (bottom).

we compare the performance of LaSR when we use logarithmic utilities (with $c_i = 100$) for all flows in the system (labeled as "log utility") and when we do not use any utility (labeled as "no utility"). The results show how we achieve high index fairness for "log utility" across all time windows, whereas "no utility" results in very poor short-term fairness. Note that "no utility" also achieves high long-term fairness because our algorithm guarantees that the (long-term) mean demand of all flows is met. We do not show results for QSG because it does not support elastic flows.

5.5 Users moving

We present next a mobility experiment with two RATs (e.g. small-cells) with identical bandwidth. Both RATs use non-interfering and partially overlapping 10-MHz bands and we set an VRB equal to one PRB. We simulate a scenario with 5 traffic classes where users move together (e.g. a train car) from the proximity of RAT 1 ($t = 0$) to the proximity of RAT 2 ($t = 12000$) presenting an aggregate traffic demand of 6.5 Mb/s. In $t = 0$ users' *best* MCS index with RAT 1 is 28 (from LTE specification [45]) and there is no coverage from RAT 2. Users move such that at $t = 12000$ we have the reverse setup (no coverage from RAT 1 and up to MCS index 28 with RAT 2). In $t = 6000$ the wireless link only allows users to use MCS index 12 from either RAT to simulate that the crowd is at cell border, that is, same channel conditions towards either RAT.

We depict in Fig. 15, for both QSG and LaSR, the mean RAT utilization (top), the mean rate provisioned per RAT (bottom, blue and red) and the mean aggregate throughput demanded by the crowd (bottom, green). Fig. 15b evinces that both algorithms satisfy the throughput requirements of the system (around 6.5 Mb/s of aggregate throughput). However, as revealed by Fig. 15a, QSG achieves this at the cost of higher RAT utilization due to its greedy nature. In contrast, LaSR gracefully balances the load between the two RATs as needed, and minimizes the allocated resources with no compromise in throughput.

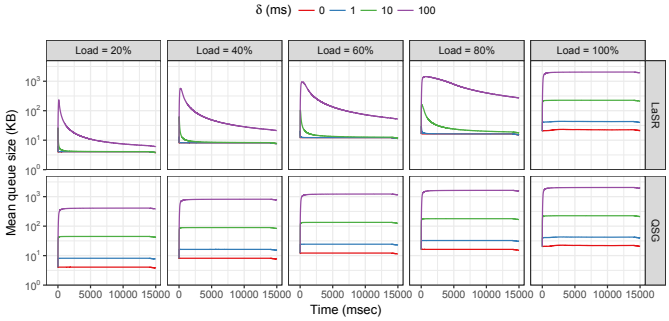


Fig. 16. Mean user queue size in a static scenario with delayed backlog information equal to δ.

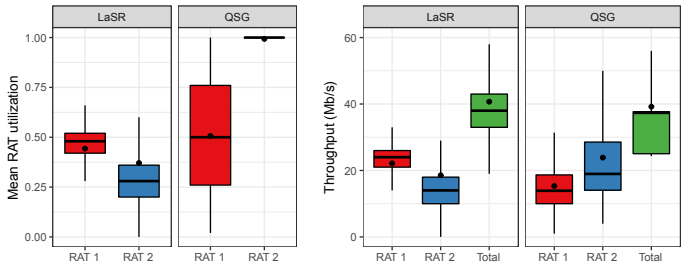
5.6 Imperfect information

We next assess the robustness of LaSR to imperfect backlog information. We set up again the same scenario we used in §5.1, with three heterogeneous RATs and different mean users loads. In this case, however, the backlog information provided to the controller suffers from some delay δ, e.g., caused by the latency of signaling this information from distributed RATs to the controller in an uplink environment. We then plot in Fig. 16 the temporal evolution of the mean queue size of each user for both LaSR and QSG and different mean loads. We observe from the figure how performance worsens as control information gets delayed and mean load increases. In addition, we can see that QSG is very sensitive to delayed information regardless of the load in the system. In contrast, LaSR barely suffers of performance degradation at low- and mid-load regimes and convergence is very quick (a few milliseconds) in most cases.

5.7 Variable and uncertain bandwidth

We now set an experiment with two RATs and 10 traffic types. RAT 1 has a bandwidth that varies every TTI uniformly at random between 0-10 MHz. This allows us to emulate a RAT operating in an unlicensed channel governed by a duty-cycle access mechanism; see LTE-U CSAT [51]. RAT 2 has fixed bandwidth of 10 MHz (e.g. a macro cell operating in a licensed band). We set an VRB equal to one PRB. The aggregate mean traffic demand is 50% of the capacity of RAT 2 (around 40 Mb/s of aggregate load). In addition, cost vector $\boldsymbol{w}$ is set so that RAT 1 (illustrating an unlicensed band) is preferred,

$$\begin{matrix}\text{RAT 1}\\ \text{RAT 2}\end{matrix}\begin{pmatrix}1\\100\end{pmatrix}\text{bit}^{-1}.$$



(a) RAT utilization (b) Throughput

Fig. 17. A cell with 2 RATs and 40 users. RAT 1 has variable bandwidth uniformly distributed between 0 and 10 MHz. RAT 2 has fixed bandwidth with 10 MHz. Box and whiskers show the relative PRB utilization of each RAT (left) and aggregate and per-RAT throughput (right). Mean values are represented with dots.

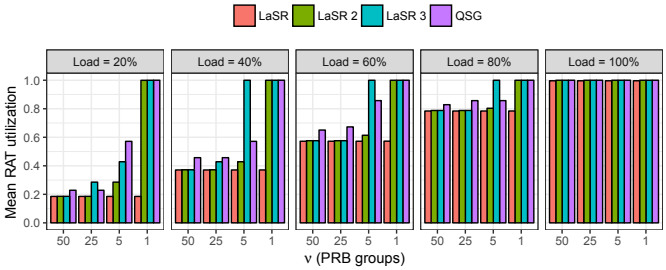


Fig. 18. Mean RAT load as a function of ν (groups of PRBs that are constrained to equal schedule choice).

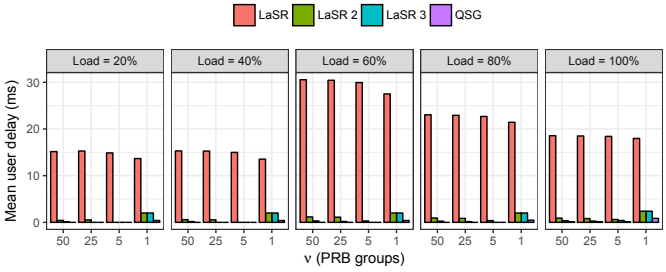


Fig. 19. Mean user delay as a function of ν (groups of PRBs that are constrained to equal schedule choice).

and users are located homogeneously with the highest modulation index for an LTE RAT (28) for both RATs. We compare in Fig. 17 the performance of both QSG and LaSR in one random instance that lasts 10 seconds (RAT 1's bandwidth oscillates every TTI). The left-hand plot depicts box and whiskers of the relative utilization of each RAT (RAT 1's results are relative to the maximum bandwidth) and the right-hand plot shows the throughput provided by each of the RATs and the aggregate ("Total"). The mean values are represented with dots. The results illustrate how LaSR is capable of satisfying the user demands with much fewer resources from RAT 2 (which simulates a licensed band) as compared to its benchmark, which, perhaps surprisingly (given that the load equals half of the capacity of that RAT), employs 100% of the resources of RAT 2 at all times. The reason lies upon the greediness of QSG, which assigns data into PRBs *even when there is not enough data to fill up a PRB*, i.e. padding is required. The figure clearly illustrates the penalty of using such greedy approaches when demand is low. On the contrary, as shown in Fig. 17a, LaSR achieves a better balance between the two RATs.

5.8 Scheduling constraints

One of the main advantages of LaSR is its ability to consider practical scheduling constraints, e.g., the way scheduling information is propagated to users (see §1). In order to illustrate this, we analyze the same scenario as in §5.1, upon different degrees of granularity in the way we can make scheduling choices (for example due to the utilization of a constrained signaling protocol). To this aim, we define ν as the amount of VRBs in our system, so that, depending on the RAT capacity, it indicates *the amount of PRBs that must have the same assignment and modulation within a TTI*. For instance, $\nu = 50$ indicates that the macro RAT (with 50 PRBs) can assign each PRB individually (maximum granularity), while the mmWave-based RAT (with 100 PRBs) is forced to make identical choices every two consecutive PRBs. On the other hand, $\nu = 1$ forces LaSR to make the same scheduling choice for all PRBs within one TTI (lowest granularity).

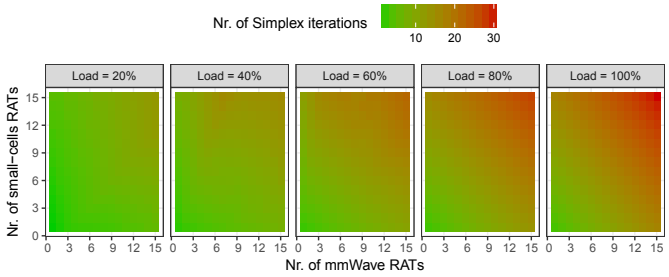


Fig. 20. Mean number of simplex iterations per VRB to solve Problem 3 as a function of the number of secondary RATs and mean load.

In Fig. 18 and Fig. 19 we compare, respectively, the utilization of physical resources and mean delay when using LaSR and the greedy scheduler, QSG, for different mean loads and ν values. In particular, the configurations labeled as "LaSR" and "LaSR2" use costs hundredfold and fourfold greater than that in Eq. 17, while "LaSR3" keeps the same configuration as in §5.1. The cost selected for configuration "LaSR" targets saving RAT resources (at the price of delay) whereas the others seek to gradually improve delay (at the price of higher physical resource usage).

The results evince that, while QSG tends to over-utilize resources for low- and mid-load ranges, "LaSR" matches better the utilization of physical resources to the mean demand when required. Take, for instance, the case with 20% of mean load (low load) and $\nu = 1$ (lowest granularity). A greedy scheduler like QSG uses 100% of physical resources because its sole goal is to minimize delay and therefore assigns data into PRBs as soon as possible. This causes that, in case of $\nu = 1$, *padding is added* in many PRBs, because they must be scheduled (due the low granularity imposed in this case) but there is no real data to be sent. In contrast, "LaSR" is able to adapt the mean demand to the actual utilization of resources at the cost of user delay. Note that, if alternatively a better delay performance is sought, we can reduce the cost of using physical resources (e.g. "LaSR2"), RAT utilization can be traded off for lower delay *with no resource wastage*.

5.9 Scalability

Finally, we assess the scalability of Algorithm 1. The most important factors in this regard are (i) the convex solver of Problem 3 (step 8) and (ii) managing the set of actions (step 10) that grows exponentially with the scenario density. To evaluate this, we set up a scenario similar to the one in §5.1, namely there is always one macro cell (primary RAT), and we vary the number of secondary RATs (small-cells and mmWave cells) between 1 and 30. This setup allows us to assess worst-case scenarios with an LTE carrier aggregation BS, where the maximum number of RATs allowed by the 3GPP specification is 32 [52], although in practice it is far below.[10] Moreover, the upcoming LTE and 5G modems for user terminals will support up to 8 concurrent RATs (e.g., Qualcomm Snapdragon X24 chipset will support 7 RATs for LTE and Snapdragon X50 will support 8 RATs for 5G).

To assess the first factor, Fig. 20 depicts the average number of iterations per VRB that are required to solve Problem 3. In the figure, the x-axis indicates the number of secondary small-cells (from 1 to 15), the y-axis shows the number of mmWave secondary RATs (from 1 to 15) and the

10. Moreover, given the larger bandwidth allocated to New Radio, the maximum number of RATs supported by 5G BSs is 16 [52].

heatmap color indicates the number of *Simplex* iterations. Note that a primary macro-cell is persistently present and we repeat the experiment for different mean aggregate user loads. Because our objective function is linear in this case, here we have adopt the *Simplex* method to solve Problem 3. As can be observed, the system load is the main factor that makes the algorithm complexity increase. In addition, the density of RATs have also a remarkable impact on the number of cycles to solve the problem. All in all, the number of iterations remain rather low, below 30, even in saturation conditions with the maximum number of RATs. Note that, for the scenarios considered in this section, one iteration takes less than 4 μs in a generic laptop with a processor Intel Xeon CPU E3-1241 v3 @ 3.50GHz. Regarding the second factor, the memory required to allocate the (sparse) matrix of available actions is 54 MB in the largest scenario shown in the figure, with a total of 31 RATs and maximum load. We can thus conclude that our approach can accommodate real-life scenarios with mild complexity.

6 Conclusions

Network densification, built upon dense and heterogeneous radio access technologies (multi-RAT systems) and multi-connectivity technology, is expected to be key to supporting the stringent requirements of next generation mobile systems (5G and beyond). The problem is that, with network densification, radio resource scheduling becomes substantially more complex. In this paper we introduced the design of LaSR (Lagrange approximation Supple Radio controller), a practical, yet effective, multi-connectivity scheduler for multi-RAT systems. LaSR is based on a stochastic subgradient method and a simple online algorithm that makes optimal discrete control actions with no prior knowledge on the user traffic patterns. As shown in the paper, the suppleness of LaSR to accommodate real-system constraints while guaranteeing system stability, and a good balance between system cost and individual utility satisfaction are its main advantages over related work. Examples of constraints evaluated in this paper include (but are not limited to): heterogeneous RATs, delays to activate/deactivate RATs, discrete sets of available modulations, constraints in the way scheduling choices can be encoded onto signaling protocols (LTE/NR's DCI), imperfect available information, or duty cycles when using unlicensed spectrum.

References

[1] J. G. Andrews, *et al.*, "What Will 5G Be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[2] N. Bhushan, *et al.*, "Network Densification: The Dominant Theme for Wireless Evolution into 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 82–89, February 2014.

[3] H. Wang, *et al.*, "Understanding Mobile Traffic Patterns of Large Scale Cellular Towers in Urban Environment," in *Proceedings of the 2015 ACM Conference on Internet Measurement Conference*, ser. IMC '15. New York, NY, USA: ACM, 2015, pp. 225–238.

[4] J. Wu, *et al.*, "Energy-efficient base-stations sleep-mode techniques in green cellular networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 17, no. 2, pp. 803–826, Secondquarter 2015.

[5] J. Huang, *et al.*, "An In-depth Study of LTE: Effect of Network Protocol and Application Behavior on Performance," *SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 363–374, Aug. 2013.

[6] B. Bangerter, *et al.*, "Networks and devices for the 5g era," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 90–96, February 2014.

[7] A. Ravanshid, *et al.*, "Multi-connectivity Functional Architectures in 5G," in *2016 IEEE International Conference on Communications Workshops (ICC)*, May 2016, pp. 187–192.

[8] 3GPP, "Technical Specification Group Radio Access Network; Physical layer; General description (Release 15)," 3rd Generation Partnership Project (3GPP), TS 38.201, December 2017.

[9] Z. Shen, *et al.*, "Overview of 3GPP LTE-advanced Carrier Aggregation for 4G Wireless Communications," *IEEE Communications Magazine*, vol. 50, no. 2, pp. 122–130, February 2012.

[10] F. Capozzi, *et al.*, "Downlink Packet Scheduling in LTE Cellular Networks: Key Design Issues and a Survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 678–700, February 2013.

[11] M. Andrews, "A Survey of Scheduling Theory in Wireless Data Networks," in *Wireless Communications*. Springer, 2007, pp. 1–17.

[12] N. Prasad, *et al.*, "Multiuser Scheduling in the 3GPP LTE Cellular Uplink," *IEEE Transactions on Mobile Computing*, vol. 13, no. 1, pp. 130–145, Jan 2014.

[13] R. Kwan, C. Leung, and J. Zhang, "Proportional fair multiuser scheduling in lte," *IEEE Signal Processing Letters*, vol. 16, no. 6, pp. 461–464, June 2009.

[14] H. Shajaiah, A. Abdel-Hadi, and C. Clancy, "Utility Proportional Fairness Resource Allocation with Carrier Aggregation in 4G-LTE," in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, Nov 2013, pp. 412–417.

[15] ——, "An efficient multi-carrier resource allocation with user discrimination framework for 5g wireless systems," *International Journal of Wireless Information Networks*, vol. 22, no. 4, pp. 345–356.

[16] A. Abdelhadi and C. Clancy, "An Optimal Resource Allocation with Joint Carrier Aggregation in 4G-LTE," in *Computing, Networking and Communications (ICNC), 2015 International Conference on*, Feb 2015, pp. 138–142.

[17] G. Yu, *et al.*, "Joint Downlink and Uplink Resource Allocation for Energy-Efficient Carrier Aggregation," *IEEE Transactions on Wireless Communications*, vol. 14, no. 6, pp. 3207–3218, June 2015.

[18] Z. L. Fazliu, C. F. Chiasserini, and G. M. Dell'Aera, "Downlink Transmit Power Setting in LTE HetNets with Carrier Aggregation," in *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–9.

[19] Y. Wang, *et al.*, "Carrier load balancing and packet scheduling for multi-carrier systems," *IEEE Transactions on Wireless Communications*, vol. 9, no. 5, pp. 1780–1789, May 2010.

[20] F. Liu, *et al.*, "Design and Performance Analysis of An Energy-Efficient Uplink Carrier Aggregation Scheme," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 2, pp. 197–207.

[21] X. Cheng, G. Gupta, and P. Mohapatra, "Joint Carrier Aggregation and Packet Scheduling in LTE-advanced Networks," in *2013 IEEE International Conference on Sensing, Communications and Networking (SECON)*, June 2013, pp. 469–477.

[22] A. Rizk and M. Fidler, "Queue-aware uplink scheduling with stochastic guarantees," *Computer Communications*, 2016.

[23] A. G. Gotsis *et al.*, "Analytical Modelling and Performance Evaluation of Realistic Time-controlled M2M Scheduling over LTE Cellular Networks," *Transactions on Emerging Telecommunications Technologies*, vol. 24, no. 4, pp. 378–388, 2013.

[24] M. J. Neely and S. Supittayapornpong, "Dynamic Markov Decision Policies for Delay Constrained Wireless Scheduling," *IEEE Transactions on Automatic Control*, vol. 58, no. 8, pp. 1948–1961.

[25] D. J. Dechene and A. Shami, "Energy efficient QoS constrained scheduler for SC-FDMA uplink," *Physical Communication*, vol. 8, pp. 81 – 90, 2013.

[26] M. Kalil, A. Shami, and A. Al-Dweik, "QoS-aware Power-efficient Scheduler for LTE Uplink," *IEEE Transactions on Mobile Computing*, vol. 14, no. 8, pp. 1672–1685, Aug 2015.

[27] L. Tassiulas and A. Ephremides, "Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, Dec 1992.

[28] S. Shakkottai and A. L. Stolyar, "Scheduling for multiple flows sharing a time-varying channel: The exponential rule," *Translations of the American Mathematical Society-Series 2*, vol. 207, pp. 185–202.

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

[30] J. Liu, *et al.*, "Heavy-ball: A new approach to tame delay and convergence in wireless network optimization," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, April 2016, pp. 1–9.

[31] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.
[32] G. Ku and J. M. Walsh, "Resource allocation and link adaptation in lte and lte advanced: A tutorial," *IEEE Communications Surveys Tutorials*, vol. 17, no. 3, pp. 1605–1633, thirdquarter 2015.
[33] A. G. Marques, *et al.*, "Optimal Cross-Layer Resource Allocation in Cellular Networks Using Channel- and Queue-State Information," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 6, pp. 2789–2807, July 2012.
[34] A. Nedic and A. Ozdaglar, "Subgradient Methods in Network Resource Allocation: Rate Analysis," in *Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on*, March 2008, pp. 1189–1194.
[35] G. D. Celik *et al.*, "Dynamic server allocation over time-varying channels with switchover delay," *IEEE Transactions on Information Theory*, vol. 58, no. 9, pp. 5856–5877, Sept 2012.
[36] M. J. Neely, E. Modiano, and C. P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 2, pp. 396–409, April 2008.
[37] M. J. Neely, "Optimal Energy and Delay Tradeoffs for Multiuser Wireless Downlinks," *IEEE Transactions on Information Theory*, vol. 53, no. 9, pp. 3095–3113, Sept 2007.
[38] V. Valls and D. J. Leith, "A Convex Optimization Approach to Discrete Optimal Control," *IEEE/ACM Transactions on Automatic Control*, vol. PP, no. 99, pp. 1–1, 2018.
[39] J. Heo, Y. Wang, and K. Chang, "A novel two-step channel-prediction technique for supporting adaptive transmission in ofdm/fdd system," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 1, pp. 188–193, Jan 2008.
[40] S. Meyn, *Control techniques for complex networks*. Cambridge University Press, 2008.
[41] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.
[42] J.-W. Lee, R. R. Mazumdar, and N. B. Shroff, "Downlink power allocation for multi-class wireless systems," *IEEE/ACM Trans. Netw.*, vol. 13, no. 4, pp. 854–867, Aug. 2005. [Online]. Available: http://dx.doi.org/10.1109/TNET.2005.852888
[43] J. Renegar, "A polynomial-time algorithm, based on newton's method, for linear programming," *Mathematical Programming*, vol. 40, no. 1, pp. 59–93, Jan 1988. [Online]. Available: https://doi.org/10.1007/BF01580724
[44] 3GPP, "Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification," 3rd Generation Partnership Project (3GPP), TS 36.331, October 2016.
[45] ——, "Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," 3rd Generation Partnership Project (3GPP), TS 136.213, September 2016.
[46] EU FP7 MiWEBA, D4.1, "Radio Resource Management for mm-wave Overlay HetNets. D4.1: System Level Simulator Specification." Tech. Rep., December 2014.
[47] N. Bui and J. Widmer, "OWL: A Reliable Online Watcher for LTE Control Channel Measurements," in *ACM All Things Cellular (MobiCom Workshop)*, Nov. 2016.
[48] I. Gomez-Miguelez, *et al.*, "srsLTE: An Open-source Platform for LTE Evolution and Experimentation," in *Proceedings of the Tenth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization*, ser. WiNTECH '16. New York, NY, USA: ACM, 2016, pp. 25–32.
[49] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A Quantitative Measure Of Fairness And Discrimination For Resource Allocation In Shared Computer Systems," DEC-TR-301, Digital Equipment Corporation, Tech. Rep., Sept. 1984.
[50] C. L. Barrett, *et al.*, "Analyzing the Short-term Fairness of IEEE 802.11 in Wireless Multi-hop Radio Networks," in *Proceedings. 10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems*, 2002, pp. 137–144.
[51] B. Chen, *et al.*, "Coexistence of LTE-LAA and Wi-Fi on 5 GHz With Corresponding Deployment Scenarios: A Survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 1, pp. 7–32, Firstquarter 2017.
[52] J. Jeon, "NR Wide Bandwidth Operations," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 42–46, March 2018.