Computer Networks 165 (2019) 106939

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Integer linear programming formulations for the variable data rate and variable channel bandwidth scheduling problem in wireless networks

José Maurício Costa, Pedro P. Paniago, Joaquim de Andrade, Thiago F. Noronha*, Marcos A.M. Vieira

Universidade Federal de Minas Gerais, Belo Horizonte, Brazil

A R T I C L E   I N F O

Article history:
Received 16 April 2019
Revised 21 September 2019
Accepted 4 October 2019
Available online 4 October 2019

Keywords:
Link scheduling
Wireless scheduling problem
Integer programming
Combinatorial optimization

A B S T R A C T

The IEEE 802.11ac standard enables a higher transmission speed than the previous IEEE standards because it allows the network frequency spectrum to be divided into communication channels of different bandwidths, varying from 20MHz to 160MHz. In this paper, we introduce the Variable Rate and Variable Bandwidth Scheduling Problem (VRBSP), which is a generalization of the classical Variable Rate Scheduling Problem (VRSP) on wireless networks. The best algorithm in the literature of VRSP, Datarate PPTAS, can only be efficiently run on networks with up to 64 links and does not guarantee optimality. In this paper, we propose two Mixed Integer Linear Programming (MILP) formulations that are used within a MILP solver to seek optimal schedules for VRBSP. This approach can also be used to solve VRSP. The computational experiments were carried out on classical network instances from the literature with up to 2048 links. They show that the MILP-based exact algorithms were able to find optimal VRBSP schedules for networks with up to 256 links and optimal VRSP schedules for networks with up to 1024 links within 3600 seconds of running time.

1. Introduction

Wireless networks are becoming ubiquitous, due to the ever increasing number of wireless devices operating in these networks. Besides, the applications that run on these devices demand an ever increasing data transmission rate. Therefore, new technologies and communication standards have being developed to deal with these issues [1,2].

A *link* in wireless communication networks can be model as a connection between a sender and a receiver device. As the transmission medium is shared, if two links transmit at the same time using overlapping bands of the electromagnetic spectrum, the sender of one of them may significantly interfere with the signal that reaches the receiver of the other. The smaller is the interference level, the larger is the signal quality that reaches the receiver. Furthermore, the larger is the signal quality, the faster is the link transmission speed. On the other hand, the interference level can be so high that it decreases the signal quality up to the point that it cannot be decoded by the receiver [3].

The latest communication standard in Wi-Fi networks is the IEEE 802.1 1ac [4]. This standard enables a higher transmission speed than the previous IEEE 802.11b and IEEE 802.11n standards, because it allows more flexibility on how the network frequency spectrum can be divided. In this case, the spectrum can be divided into communication channels with bandwidths, varying from 20 MHz to 160 MHz. For example, a single band of 160 MHz of the electromagnetic spectrum can be partitioned into 8 channels of 20 MHz, 4 channels of 40 MHz, 2 channels of 80 MHz, or a single channel of 160 MHz.

The larger flexibility allowed by the IEEE 802.1 1ac standard brings another layer of decision making. Therefore, in this paper, we generalize the classical Variable Rate Scheduling Problem (VRSP) [5,6], and introduce a new problem called Variable Rate and Variable Bandwidth Scheduling Problem (VRBSP). As VRSP is a special case of VRBSP, where all channels have the same bandwidth, VRBSP is at least as hard to solve than VRSP. Therefore, as VRSP is NP-hard [5,6], so is VRBSP. As there is no known approach to design polynomial-time exact algorithms for problems in this class, this paper contributes to the literature by proposing two Mixed Integer Linear Programming (MILP) formulations for VRBSP. They are solved by a state-of-the-art MILP solver to seek optimal schedules for VRBSP. The computational experiments showed that our MILP-based exact algorithms were able to find optimal VRBSP schedules

* Corresponding author.
E-mail addresses: jmauricio@dcc.ufmg.br (J.M. Costa), paniago@dcc.ufmg.br (P.P. Paniago), joaquim.andrade@dcc.ufmg.br (J. de Andrade), tfn@dcc.ufmg.br (T.F. Noronha), mmvieira@dcc.ufmg.br (M.A.M. Vieira).

https://doi.org/10.1016/j.comnet.2019.106939
1389-1286/

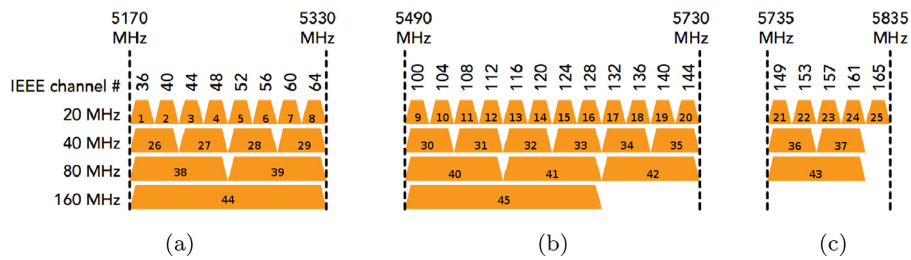


Fig. 1. The subdivisions of the electromagnetic spectrum into communication channels defined by the IEEE 802.11ac standard [7].

for networks with up to 256 links and optimal VRSP schedules for networks with up to 1024 links within 3600 seconds of running time. Within this time limit, the best algorithm in the literature of VRSP, Datarate PPTAS [6], could only provide feasible schedules for networks with up to 64 links, none of which were optimal.

The remainder of this paper is organized as follows. VRBSP is formally defined in Section 2, and related work is discussed in Section 3. Next, the two Mixed Integer Linear Programming (MILP) formulations for VRBSP are presented in Section 4. Then, the computational experiments are reported in Section 5, and concluding remarks are drawn in the last section.

2. Problem definition

Let $V = \{1, \ldots, n\}$ be the set of network devices, and $L \subseteq V \times V$ be the set of links, such that a link $(i, j) \in L$ has a *sender* device $i \in V$ and a *receiver* device $j \in V$. Two links might share a device, but in this case these links cannot transmit simultaneously. Let also B be the set of channel bandwidths (in MHz). In the case of the IEEE 802.11ac standard, we have that $B = \{20, 40, 80, 160\}$ [2]. Besides, let C be a set of subdivisions of the electromagnetic spectrum into communication channels, whose bandwidth is one of those in B. These channels might overlap with each other. Therefore, we also define a set $O^c \subseteq C$ with the channels that overlap with $c \in C$ (including itself). In the case of the IEEE 802.11ac standard, we have that $C = \{1, \ldots, 45\}$ and O^c, for all $c \in C$, is given by Fig. 1 [7]. For example, $O^{32} = \{13, 14, 32, 41, 45\}$ because channel 32 overlaps with channels 13, 14, 41, 45, and itself. In addition, we also define $C^b \subseteq C$ as the subset of channels whose bandwidth is $b \in B$, and $B^c \in B$ as the bandwidth of channel $c \in C$. For example, we have that $B^{44} = B^{45} = 160$. As they are the only channels with a bandwidth of 160 MHz, we also have that $C^{160} = \{44, 45\}$.

The Variable Rate and Variable Bandwidth Scheduling Problem (VRBSP) consists in selecting a subset of the links in L, that will transmit in the same time slot, and assign a channel in C to each of them, such that (i) no two links that share a device are selected together, and (ii) the interference at the receiver of each scheduled link is small enough so that the transmitted message can be decoded. A solution for VRBSP is represented by a schedule $S \subseteq L \times C$, where $\langle (i, j), c \rangle \in S$ denotes that link $(i, j) \in L$ is scheduled to channel $c \in C$. For the sake of a clearer notation, we also denote by $L(S)$ the set of *scheduled links*, i.e., $L(S) = \{(i, j) \in L : \exists \langle (i, j), c \rangle \in S\}$. The objective of VRBSP is to find the schedule with the maximum network *throughput*, i.e., the schedule that maximizes the sum of the data transmission rate of the scheduled links.

We adopted the Signal to Interference plus Noise Ratio (SINR) model [8] to represent the quality of the signal that reaches the receiver of each link $\langle (i, j), c \rangle \in S$. This value is denoted by $SINR_{ij}$ and, together with the value of B^c, determines the data transmission rate of (i, j). The value of $SINR_{ij}$ is inversely proportional to the interference I_{ij} at the receiver of (i, j), caused by the other scheduled links that are assigned to channels that overlap with c. As in [6,9,10], we assume that there is no interference between links assigned to non-overlapping channels, and that all sender devices transmit with the same power level P (in Watts). Thus, the value of $SINR_{ij}$ (in Decibels) in a schedule S can be computed by Eq. (1) [8],

$$SINR_{ij} = \frac{\frac{P}{(d_{ij})^{\alpha}}}{I_{ij} + N}, \forall \langle (i, j), c \rangle \in S \tag{1}$$

where d_{ij} is the distance between $i \in V$ and $j \in V$ (in meters), $\alpha \in \mathbb{R}$ is the path-loss exponent [11], with typical values in the range $2 < \alpha \leq 6$ [3,8,9], N is the ambient noise (in Watts) [8], with typical values in the range $0 \leq N \leq 8 \cdot 10^{-14}$ [3,6,12,13], and $I_{ij} = \sum_{\langle (u,v), c' \rangle \in S \setminus \{\langle (i,j), c \rangle\} : c' \in O^c} \frac{P}{(d_{uj})^{\alpha}}$.

In the IEEE 802.11ac standard, the data transmission rate of a scheduled link $\langle (i, j), c \rangle \in S$, denoted by r_{ij}, is computed by mapping $SINR_{ij}$ and B^c to the *Modulation and Coding Scheme* (MCS) [1] that results in largest possible data transmission rate. The IEEE 802.11ac standard has 10 MCS [2]. Let $M = \{0, \ldots, 9\}$ be the set of MCS identifiers, we denote by q^b_m the minimum SINR that a link must have to be able to transmit with a data rate of $\bar{r}^b_m$ using a channel with bandwidth $b \in B$ and the MCS $m \in M$. The values of q^b_m and $\bar{r}^b_m$, for all $b \in B$ and $m \in M$, defined by the IEEE 802.11ac standard, are given by Table 1. Column 1 identifies the MCS, and Column 2 gives the minimum value of SINR q^b_m necessary to transmit with a date rate of $\bar{r}^b_m$ (shown in Column 3) using a channel of bandwidth $b = 20$ and the MCS $m \in M$. The last line in columns 2 and 3 is filled with '–', because the largest data rate a link can transmit in a channel of bandwidth 20 MHz is 78.0 Mbps. The same data is reported for channels of bandwidth 40 MHz in columns 4 and 5, of bandwidth 80 MHz in columns 6 and 7, and of bandwidth 160 MHz in columns 8 and 9, respectively. For example, according to Table 1, a link $\langle (i, j), c \rangle \in S$ that is assigned to a channel of 80 MHz and that has $SINR_{ij} = 22.0$ dB transmits at a data rate of 175.5 Mbps. If $SINR_{ij} \geq 37.0$ dB the same link would transmit at a data rate of 390.0 Mbps. However, if $SINR_{ij} < 8.0$ dB the receiver of this link would not be able to decode the message, and this link could not be scheduled. Thus, given $SINR_{ij}$ and B^c, the transmission data rate of $\langle (i, j), c \rangle \in S$ is given by Eq. (2).

$$r_{ij} = \max_{m \in M : SINR_{ij} \geq q^{B^c}_m} \bar{r}^{B^c}_m \tag{2}$$

Therefore, VRBSP can be formally defined by Eq. (3), where $\Delta \subset 2^{L \times C}$ is the set of all *feasible schedules*, i.e., those schedules $S \in \Delta$ where (i) no two links in S share a device and (ii) $SINR_{ij} \geq \min_{m \in M} q^{B^c}_m$ for all $\langle (i, j), c \rangle \in S$.

$$S^* = \operatorname{argmax}_{S \in \Delta} \sum_{(i,j) \in L(S)} r_{ij} \tag{3}$$

3. Related work

Gupta and Kumar [8] formalized two interference models in wireless networks: the protocol model and the physical interference model (also known as the Signal to Interference plus Noise Ratio – SINR). The computational experiments showed that the SINR model is more accurate than the protocol model. Therefore,

Table 1
SINR(dB) and Data Rate (Mbps) for each channel bandwidth according to the IEEE 802.1 1ac standard.

MCS Index	20 MHz		40 MHz		80 MHz		160 MHz	
	SINR q_m^{20}	Data rate $\bar{r}_m^{20}$	SINR q_m^{40}	Data rate $\bar{r}_m^{40}$	SINR q_m^{80}	Data rate $\bar{r}_m^{80}$	SINR q_m^{160}	Data rate $\bar{r}_m^{160}$
0	2.0	6.5	5.0	13.5	8.0	29.3	11.0	58.5
1	5.0	13.0	8.0	27.0	11.0	58.5	14.0	117.0
2	9.0	19.5	12.0	40.5	15.0	87.8	18.0	175.5
3	11.0	26.0	14.0	54.0	17.0	117.0	20.0	234.0
4	15.0	39.0	18.0	81.0	21.0	175.5	24.0	351.0
5	18.0	52.0	21.0	108.0	24.0	234.0	27.0	468.0
6	20.0	58.5	23.0	121.5	26.0	263.3	29.0	526.5
7	25.0	65.0	28.0	135.0	31.0	292.5	34.0	585.0
8	29.0	78.0	32.0	162.0	35.0	351.0	38.0	702.0
9	–	–	34.0	180.0	37.0	390.0	40.0	780.0

this work and the related work discussed below focus on the SINR model.

3.1. Single rate problems

Ephremides and Truong [14] proposed the first problem of scheduling links in wireless networks where the signal quality is represented by the SINR model. In this problem, all links transmit at the same data rate and share the same communication channel. The Wireless Scheduling Problem (WSP) consists in partitioning the set of links L into the smallest possible number of subsets, so that $SINR_{ij} \geq \beta$, for every link $(i, j) \in L$, where β is the minimum threshold required for the signal to be decoded. The links in the same subset are scheduled in the same time interval. A proof that WSP is NP-hard was presented in [15].

Early work on this problem focused on approximate algorithms. Brar et al. [16] proposed the GreedyPhysical algorithm, whose approximation factor is $O(n^{1-\frac{2}{\psi(\alpha)+\epsilon}} \cdot (log(n))^{\frac{2}{\psi(\alpha)+\epsilon}})$, where α is the attenuation factor, $\psi(\alpha) = \frac{1}{2} + \frac{\sqrt{9\alpha-20\alpha+4}}{2(\alpha-2)}$, and $\epsilon > 0$ is an arbitrarily small value. Computational experiments have shown that GreedyPhysical obtains solutions up to three times better than those of the approach proposed in [17] that uses simulation. The ApproxDiversity algorithm was proposed in [15]. It has an approximation factor of $O(\log \frac{d^{max}}{d^{min}})$, where d^{max} and d^{min} are the longest and the shortest distance between the transmitter and the receiver of a link, respectively. Gousevskaia et al. [9] proposed another algorithm, called ApproxLogN, whose approximation factor is $O(\log n)$. The computational experiments performed in this work, on networks with up to 12800 links, showed that ApproxLogN had a better performance than GreedyPhysical [16] and ApproxDiversity [15] on average.

Djukic and Valaee [18] proposed a variation of WSP, where each link occupies one or more contiguous time intervals. In this work, the authors proposed a heuristic that divides the problem into two parts. First, the order in which each link is scheduled is decided. Then, it is decided at what time interval each link begins to transmit. The first subproblem is modeled by an integer linear programming formulation, and solved optimally using a branch-and-bound algorithm. In addition, the authors show that, given any order for the links, the second subproblem can be solved by applying the Bellman-Ford algorithm [19,20] in a conflict graph. Computational experiments have shown that the computational time required to solve the first subproblem is prohibitive. However, this subproblem can be solved in polynomial time in networks whose topology is a tree. Chilukuri and Sahoo [21] extended this technique and proposed a heuristic to solve the first subproblem. The second subproblem is solved in the same way as in [18]. In addition to being efficient for other network topologies, computational experiments in networks with up to 10 links have shown that this new approach finds solutions as good as those obtained by the heuristics in [18].

Vieira et al. [13] proposed a heuristic to solve a variation of WSP in which each link is allocated to exactly τ distinct time slots. At each iteration, the heuristic selects a link from L and inserts it in the first interval of time where it and the other links already allocated to that interval of time can transmit simultaneously. If the link cannot be inserted in any of the time intervals already used, the link is allocated to a new time interval. This procedure is repeated τ times, so that at the end, each link is allocated at exactly τ time intervals. Computational experiments on networks with up to 12800 links showed that the proposed heuristic was able to find better solutions on average than GreedyPhysical [16] and ApproxLogN [9].

Goussevskaia et al. [15] proposed the Single-Shot Scheduling Problem (SSSP). This problem consists in scheduling a subset $S \subseteq L$ of links in a single time interval, so that the number of links that transmit in this interval is maximized and that $SINR_{ij} \geq \beta$, for all $(i, j) \in S$. A proof that SSSP is NP-hard was also presented in [15].

In Goussevskaia et al. [15], an algorithm for SSSP was developed whose approximation factor is $O(\log \frac{d^{max}}{d^{min}})$. Already in [9] an algorithm has been proposed for the same problem with approximation factor $O((2 \cdot c + 1)^{\alpha} + (3^{\alpha+1} \cdot 5) + 1)$, where $c = \max\{(2, (2^5 \cdot 3^2 \cdot \beta \cdot (\frac{\alpha-1}{\alpha-2}))^{\frac{1}{\alpha}}\}$. In [22] a variation of SSSP has been studied where the transmitters can have different transmission power. They proposed an algorithm whose approximation factor is $O(\log \frac{d^{max}}{d^{min}})$. Another algorithm for this variation of SSSP was proposed by Halldórsson and Mitra [23] whose approximation factor is $O(\log |L| + \log\log \frac{d^{max}}{d^{min}})$. Kesselheim [24] proposed the first algorithm for SSSP whose approximation factor $O(\log |L|)$ varies according to the number of links only.

3.2. Multi-rate problems

Goussevskaia et al. [3] proposed the Multi-Rate Scheduling Problem (MRSP) in Wireless Networks. In this problem, the data rate can be different from one link to another. Therefore, each link $(i, j) \in L$ has a requirement β_{ij} for signal quality. MRSP consists in scheduling a subset $S \subseteq L$ of links in a single time interval, so that the network throughput in this interval is maximized and that $SINR_{ij} \geq \beta_{ij}$, for all $(i, j) \in S$. This problem is a generalization of SSSP [15], and consequently it is also NP-hard.

Goussevskaia et al. [3] proposed a Polynomial-Time Approximation Scheme (PTAS) [25], called DISK-MRS. The approximation factor of this algorithm is $(1 + \frac{1}{K-1})^2$, where $K > 1$ is a constant. Computational experiments, in networks with up to 2048 links, showed that DISK-MRS obtains better results than an adaptation of AproxDiversity [9,15] to MRSP.

Halldórsson and Mitra [12] proposed a heuristic based on an integer linear programming formulation for the MRSP. This heuristic consists of (i) solving the linear relaxation of the proposed formulation, (ii) rounding the fractional variables and (iii) constructing a viable integer solution for the problem. Computational experiments in networks with up to 600 links have shown that the proposed algorithm obtains equivalent or better results than an adaptation of the heuristic of [23] to MRSP.

3.3. Variable rate problems

Kesselheim [5] proposed the Variable Rate Scheduling Problem (VRSP) in Wireless Networks. This problem consists in selecting a single subset of links $S \subseteq L$ so that the network throughput is maximized and that $SINR_{ij} \geq \beta(r_{ij})$ for every link $(i, j) \in S$, where R is the set of data rates at which links can transmit, and $\beta(r_{ij})$ is the minimum signal quality required to transmit at data rate $r_{ij} \in R$. When $|R| = 1$, VRSP reduces to SSSP. Consequently, it is also NP-hard.

Kesselheim [5] proposed an algorithm for VRSP whose approximation factor is $O(\log |L|)$. Ásgeirsson et al. [26] proposed a framework to solve several scheduling problems in wireless networks. This framework, when applied to VRSP, results in an algorithm whose approximation factor is $O(\log \log \frac{r^{max}}{r^{min}})$, where $r^{max} = \max_{r' \in R} r'$ and $r^{min} = \min_{r' \in R} r'$.

Goussevskaia et al. [6] showed that VRSP can be modeled as a conflict graph and proposed a polynomial approximation scheme. This algorithm, called Datarate PPTAS, is an extension of the algorithm DISK-MRS [3]. The approximation factor of this algorithm is $(1 + \frac{1}{K-1})^2$, where $K > 1$ is an integer value. However, the computational complexity of this algorithm is $n^{O(K^2)}$, which makes its execution impractical for medium and large networks. Computational experiments in networks with up to 32 links have shown that Datarate PPTAS obtains better results than adaptations of AproxDiversity [9,15] and DISK-MRS [3] for VRSP.

The problem of scheduling links in wireless networks with variable data rate and variable channel bandwidth (VRBSP), defined in Section 2, is a generalization of VRSP. When $|B| = |C| = 1$, VRBSP reduces to VRSP. Therefore, it is also NP-hard. To the best of our knowledge, there is no work in the literature that has studied and evaluated the application of exact algorithms for this version of the problem that considers channels with variable bandwidth. Thus, the present work proposes the first MILP formulations for this problem, and evaluates how efficient the MILP-based exact algorithms are to solve VRBSP and its special case VRSP.

4. MILP formulations

This section presents two mixed integer linear formulations for VRBSP. In Section 4.1, we show a mixed integer nonlinear formulation, as the relationship between the interference and the SINR of a link is intrinsically nonlinear. Then, in Section 4.2, we describe two approaches to linearize this formulation.

4.1. Nonlinear formulation

Given an instance of VRBSP characterized by the tuple $\langle V, L, B, C, O^c, C^b, M, q_s^b, \bar{r}_s^b, P, d_{ij}, \alpha, N \rangle$ as defined in Section 2, VRBSP can be modeled by variables $x_{ij}^c \in \{0, 1\}$, where $x_{ij}^c = 1$ if link $(i, j) \in L$ is scheduled and assigned to channel $c \in C$, and $x_{ij}^c = 0$ otherwise. We also define auxiliary variables $y_{ij}^{bm} \in \{0, 1\}$, for all $(i, j) \in L$, $b \in B$, and $m \in M$, such that $y_{ij}^{bm} = 1$ if the data transmission rate of (i, j) is $\bar{r}_m^b$, and $y_{ij}^{bm} = 0$ otherwise. Besides, we also use auxiliary variables $I_{ij} \in \mathbb{R}$, for all $(i, j) \in L$, that give the interference at j. Thus, a nonlinear formulation for VRBSP is given by (4)–(9).

$$\text{maximize } F(x) = \sum_{(i,j)\in L} \sum_{b\in B} \sum_{m\in M} \bar{r}_m^b \cdot y_{ij}^{bm} \tag{4}$$

$$\text{subject to } \sum_{(i,j)\in L} \sum_{c\in C} x_{ij}^c + \sum_{(j,i)\in L} \sum_{c\in C} x_{ji}^c \leq 1, \ \forall i \in V \tag{5}$$

$$\sum_{m\in M} y_{ij}^{bm} \leq \sum_{c\in C^b} x_{ij}^c, \ \forall (i, j) \in L, b \in B \tag{6}$$

$$I_{ij} = \sum_{(u,v)\in L\setminus\{(i,j)\}} \frac{P}{(d_{uj})^\alpha} \cdot \left(\sum_{c\in C} \sum_{\bar{c}\in O^c} x_{ij}^c \cdot x_{uv}^{\bar{c}} \right), \ \forall (i, j) \in L \tag{7}$$

$$\frac{\frac{P}{(d_{ij})^\alpha}}{I_{ij} + N} \geq \sum_{b\in B} \sum_{m\in M} q_m^b \cdot y_{ij}^{bm}, \ \forall (i, j) \in L \tag{8}$$

$$y_{ij}^{bm} \in \{0, 1\}, \quad \forall \ (i, j) \in L, \ \forall \ b \in B, \ \forall \ m \in M \tag{9}$$

$$x_{ij}^c \in \{0, 1\}, \quad \forall \ (i, j) \in L, \ \forall \ c \in C \tag{10}$$

The objective function (4) maximizes the network throughput, i.e., the sum of the data transmission rate of the scheduled links. The constraints in (5) guarantee that every link is assigned to at most one channel, and that two links that share the same device are not scheduled together. The constraints in (6) associate the value of variables y and x and enforce that a link transmits in at most one specific data rate. In addition, the constraints in (7) compute the value of the interference I_{ij}, for all $(i, j) \in L$, while those in (8) impose that (i, j) can only transmit at a data rate of $\bar{r}_m^b$ if $SINR_{ij} \geq q_m^b$, where $SINR_{ij} = \frac{\frac{P}{(d_{ij})^\alpha}}{I_{ij}+N}$. The integrality of variables y and x is ensured by (9) and (10), respectively.

4.2. Linear formulations

The constraints in (7) and (8) are clearly nonlinear. In order to obtain a linear formulation for VRBSP, we first rewrite (8) as (11), which is linear. The latter is based on the observation that Eq. (1) can be rewritten as $I_{ij} = \frac{\frac{P}{(d_{ij})^\alpha}}{SINR_{ij}} - N$.

$$\sum_{b\in B} \sum_{m\in M} \left(\frac{\frac{P}{(d_{ij})^\alpha}}{q_m^b} - N \right) \cdot y_{ij}^{bm} \geq I_{ij}, \quad \forall (i, j) \in L \tag{11}$$

Then, we propose two approaches to linearize the constraints in (7), which compute the value I_{ij} for all $(i, j) \in L$. Both approaches make use of variables $z_{uv}^c \in \mathbb{R}$, for all $(u, v) \in L$ and $c \in C$, such that $z_{uv}^c = 1$ if $(u, v) \in L$ is assigned to a channel $\bar{c} \in O^c$ that overlaps with c, and $z_{uv}^c = 0$ otherwise. The correct value of these variables is ensured by the constraints in (12).

$$z_{uv}^c = \sum_{\bar{c}\in O^c} x_{uv}^{\bar{c}m}, \quad \forall (u, v) \in L, \forall c \in C, \forall m \in M \tag{12}$$

The first approach to linearize (7) is based on the work of [27]. In addition to (12), it makes use of variables $I_{ij}^c \in \mathbb{R}$, that give which would be the value of I_{ij} if $(i, j) \in L$ were assigned to channel $c \in C$. The correct value of these variables is ensured by the constraints in (13).

$$I_{ij}^c = \sum_{(u,v)\in L\setminus\{(i,j)\}} \frac{P}{(d_{uj})^\alpha} \cdot z_{uv}^c, \quad \forall (u, v) \in L, \forall c \in C \tag{13}$$

Thus, the value of I_{ij} can be linearly computed by (14) and (15), where the constant $M_{ij} = \sum_{(u,v)\in L\setminus\{(i,j)\}} \frac{P}{(d_{uj})^\alpha}$ is an upper bound to the value of I_{ij}^c. Therefore, VRBSP is linearly formulated by (4)-(5),(9)-(15).

$$I_{ij} \geq I_{ij}^c - M_{ij} \cdot \left(1 - x_{ij}^c\right), \quad \forall (i, j) \in L, \forall c \in C \tag{14}$$

$$I_{ij} \leq I_{ij}^c + M_{ij} \cdot \left(1 - x_{ij}^c\right), \quad \forall (i, j) \in L, \forall c \in C \tag{15}$$

The second approach to linearize (7) is based on the work of [28]. In addition to (12), this approach makes use of variables $w_{ij}^{uv} \in [0, 1]$, for all $(i, j) \in L$ and $(u, v) \in L \backslash \{(i, j)\}$, such that $w_{ij}^{uv} = 1$ if the links (i, j) and (u, v) are assigned to overlapping channels, and $w_{ij}^{uv} = 0$ otherwise. The correct value of these variables is ensured by the constraints in (16) and (17).

$$w_{ij}^{uv} \geq x_{ij}^{c} + z_{uv}^{c} - 1, \quad \forall (i, j) \in L, \forall (u, v) \in L \setminus \{(i, j)\}, \forall c \in C \quad (16)$$

$$w_{ij}^{uv} \in [0, 1], \quad \forall (i, j) \in L, \forall (u, v) \in L \setminus \{(i, j)\}, \quad (17)$$

Thus, the value of I_{ij} can be linearly computed by (18). Therefore, VRBSP is linearly formulated by (4)–(5),(9)–(12),(16)–(18).

$$I_{ij} = \sum_{(u,v) \in L \setminus \{(i,j)\}} \frac{P}{(d_{uj})^{\alpha}} \cdot w_{ij}^{uv}, \quad \forall (i, j) \in L \quad (18)$$

5. Computational experiments

In this section, we describe the computational experiments performed with the aim to evaluate and compare the exact algorithms based on the MILP formulations (4)–(5),(9)–(15) (referenced as F1) and (4)–(5),(9)–(12),(16)–(18) (referenced as F2), as well as the Datarate PPTAS algorithm of [6]. Formulations F1 and F2 were solved by the branch and cut algorithm of the IBM/ILOG CPLEX solver, version 12.6, running with the default parameters settings. We refer to B&C-F1 and B&C-F2 as the CPLEX's branch and cut algorithm based on formulations F1 and F2, respectively. The code of Datarate PPTAS was provided by the authors of [6]. It was implemented in the Java language and executed through the Java Runtime Environment (JRE) 1.8.0. The computational experiments were carried out on a machine with an Intel Core i7 3.33 GHz CPU and 24 GB of RAM memory, running the Linux operating system Ubuntu 14.04 LTS.

In this work, two different sets of instances were generated. The first set, called *D10000*, was generated exactly as in [6]. It consists of instances that have planes with dimensions equal to $10,000 \times 10,000$ meters. The set L of links is generated as follows: first, $|L|$ receivers are randomly positioned in the plane; then, for each receiver, a sender is randomly positioned within $l_{max} = 6\sqrt{2}$ meters from the receiver. The distance between the devices in the network is considered to be Euclidean. Besides, the values of parameters N, α, and P were set to 0.0, 3.0 and 1000.0, respectively, while the value of $|L|$ was varied by 8, 16, 32, 64, 128, 256, 512, 1024, and 2048 links. For each value of $|L|$, we randomly generated 30 instances. Therefore, the set *D10000* is composed of 270 instances.

The same methodology was used to generate a second set of instances, called *D250*. However, the latter is generated in a plane of 250×250 meters. Thus, the devices are more densely packed. Therefore, these instances are expected to be harder to solve than those in *D10000*. Again, for each value of $|L|$, we randomly generated 30 instances. Hence, this set has also 270 instances.

5.1. VRSP

In this section, we report the experiments to evaluate the ability of B&C-F1, B&C-F2, and Datarate PPTAS [6] to obtain solutions for VRSP. Two variants of VRSP are tackled in [6], one corresponding to the IEEE 802.11b standard and the other corresponding to the IEEE 802.11n standard. In both variants, there is only one channel available, i.e., $|C| = 1$. Besides, in the first VRSP variant, the set of bandwidths is $B = \{20\}$, and the values of q_m^{20} and $\bar{r}_m^{20}$, for all $m \in M = \{0, 1, 2, 3\}$, are given by Table 2. In the second variant, we have $B = \{40\}$, and the values of q_m^{40} and $\bar{r}_m^{40}$, for all $m \in M = \{0, \ldots, 7\}$, are given by Table 3.

Table 2
SINR(dB) and Data Rate (Mbps) for IEEE 802.11b standard.

MCS Index	20 MHz	
	SINR q_m^{20}	Data rate $\bar{r}_{ij}^{20}$
0	4.0	1.0
1	6.0	2.0
2	8.0	5.5
3	10.0	11.0

Table 3
SINR(dB) and data rate (Mbps) for IEEE 802.11n standard.

MCS Index	40 MHz	
	SINR q_m^{40}	Data rate $\bar{r}_{ij}^{40}$
0	14.0	30.0
1	17.0	60.0
2	19.0	90.0
3	22.0	120.0
4	26.0	180.0
5	30.0	240.0
6	31.0	270.0
7	32.0	300.0

The results of B&C-F1, B&C-F2, and Datarate PPTAS for the VRSP variant corresponding to the IEEE 802.11b standard are reported in Tables 4 and 5, with Table 4 displaying the results for the instances in the set *D10000* and Table 5 showing the results for the instances in set *D250*. The same results for the VRSP variant corresponding to the IEEE 802.11n standard are reported in Tables 6 and 7, with Table 6 displaying the results for the instances in set *D10000* and Table 7 showing the results for the instances in set *D250*. These four tables have the same structure. Column 1 depicts the values of $|L|$, while Column 2 gives the number of instances (among the 30 instances in each group) for which Datarate PPTAS was able to find optimal solutions. Let lb_i, for $i = 1, \ldots, 30$, be the throughput of the solution returned by Datarate PPTAS on the i-th instance, which is a lower bound to the value of the optimal solution. Column 3 displays the average of lb_i, over the 30 instances, while Column 4 shows the average of the relative optimality *gap* $(\frac{\widehat{ub}_i - lb_i}{lb_i})$. As no upper bound is computed by Datarate PPTAS, we set $\widehat{ub}_i$ as the best upper bound among those obtained by B&C-F1 and B&C-F2 for the instance i. In addition, the average, over the 30 instances, of the running times of Datarate PPTAS is reported in Column 5. Similarly, the number of instances solved to optimality by B&C-F1, within 3600 seconds, is shown in Column 6. The average, over the 30 instances, of the upper bounds ub_i and the lower bounds lb_i found by B&C-F1 are reported in columns 7 and 8. Moreover, columns 9 and 10 display the average, over the 30 instances, of the corresponding relative optimality gaps $(\frac{ub_i - lb_i}{lb_i})$ and the average running times of B&C-F1, respectively. The same results are presented for B&C-F2 in the last five columns, respectively. When one of these algorithms could not find a non-empty feasible solution within 3600 seconds, the corresponding column is filled with a dash '–'. Besides, when there was not enough memory to run any of these algorithms, the corresponding column is filled with an $\times$.

Regarding the VRSP variant corresponding to the IEEE 802.11b standard, one can see from Tables 4 and 5 that Datarate PPTAS was able to find feasible solutions, within 3600 seconds, for instances with up to 64 links in set *D10000* (as was the case in [6]), and for no instance in set *D250*. Meanwhile, B&C-F1 found optimal solutions for all 270 instances in set *D10000*, and for all instances

Table 4
Results of Datarate PPTAS, B&C-F1, and B&C-F2 on set *D10000* regarding the VRSP variant corresponding to the IEEE 802.11*b* standard.

	Datarate PPTAS				B&C-F1					B&C-F2				
$\|L\|$	#opt	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)
8	0	83.63	5.64	0.02	30	88.00	88.00	0.00	0.01	30	88.00	88.00	0.00	0.01
16	0	151.73	16.47	0.02	30	176.00	176.00	0.00	0.01	30	176.00	176.00	0.00	0.02
32	0	280.03	26.04	0.02	30	352.00	352.00	0.00	0.01	30	352.00	352.00	0.00	0.08
64	0	541.63	30.04	366.00	30	703.27	703.27	0.00	0.03	30	703.27	703.27	0.00	6.90
128	0	-	-	3600.00	30	1407.82	1407.82	0.00	0.11	×	×	×	×	×
256	0	-	-	3600.00	30	2813.08	2813.08	0.00	0.42	×	×	×	×	×
512	0	-	-	3600.00	30	5617.95	5617.95	0.00	3.17	×	×	×	×	×
1024	0	-	-	3600.00	30	11203.22	11203.22	0.00	25.33	×	×	×	×	×
2048	0	-	-	3600.00	30	22286.73	22286.73	0.00	179.22	×	×	×	×	×

Table 5
Results of Datarate PPTAS, B&C-F1, and B&C-F2 on set *D250* regarding the VRSP variant corresponding to the IEEE 802.11*b* standard.

	Datarate PPTAS				B&C-F1					B&C-F2				
$\|L\|$	#opt	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)
8	0	-	-	3600.00	30	81.40	81.40	0.00	0.04	30	81.40	81.40	0.00	0.04
16	0	-	-	3600.00	30	157.40	157.40	0.00	0.05	30	157.40	157.40	0.00	0.09
32	0	-	-	3600.00	30	277.95	277.95	0.00	0.26	30	277.95	277.95	0.00	0.68
64	0	-	-	3600.00	30	438.88	438.88	0.00	7.73	30	438.88	438.88	0.00	51.03
128	0	-	-	3600.00	13	618.72	610.37	1.38	2756.23	×	×	×	×	×
256	0	-	-	3600.00	0	911.04	748.10	21.81	3600.00	×	×	×	×	×
512	0	-	-	3600.00	0	1278.06	838.22	52.52	3600.00	×	×	×	×	×
1024	0	-	-	3600.00	0	1767.34	636.92	192.72	3600.00	×	×	×	×	×
2048	0	-	-	3600.00	0	16686.11	-	-	3600.00	×	×	×	×	×

Table 6
Results of Datarate PPTAS, B&C-F1, and B&C-F2 on set *D10000* regarding the VRSP variant corresponding to the IEEE 802.11*n* standard.

	Datarate PPTAS				B&C-F1					B&C-F2				
$\|L\|$	#opt	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)
8	0	2323.00	3.55	9.43	30	2400.00	2400.00	0.00	0.01	30	2400.00	2400.00	0.00	0.01
16	0	4238.00	13.84	94.52	30	4798.00	4798.00	0.00	0.01	30	4798.00	4798.00	0.00	0.02
32	0	7751.00	24.11	196.38	30	9582.00	9582.00	0.00	0.02	30	9582.00	9582.00	0.00	0.09
64	0	-	-	3600.00	30	19106.00	19106.00	0.00	0.05	30	19106.00	19106.00	0.00	5.05
128	0	-	-	3600.00	30	37949.00	37949.00	0.00	0.19	×	×	×	×	×
256	0	-	-	3600.00	30	75014.00	75014.00	0.00	1.20	×	×	×	×	×
512	0	-	-	3600.00	30	146739.00	146739.00	0.00	20.28	×	×	×	×	×
1024	0	-	-	3600.00	30	278334.00	278334.00	0.00	991.90	×	×	×	×	×
2048	0	-	-	3600.00	0	503874.69	203126.00	436.01	3600.00	×	×	×	×	×

Table 7
Results of Datarate PPTAS, B&C-F1, and B&C-F2 on set *D250* regarding the VRSP variant corresponding to the IEEE 802.11*n* standard.

	Datarate PPTAS				B&C-F1					B&C-F2				
$\|L\|$	#opt	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)
8	0	-	-	3600.00	30	1226.00	1226.00	0.00	0.25	30	1226.00	1226.00	0.00	0.33
16	0	-	-	3600.00	30	1619.00	1619.00	0.00	2.48	30	1619.00	1619.00	0.00	5.65
32	0	-	-	3600.00	30	1934.00	1934.00	0.00	146.66	29	1937.70	1934.00	0.19	1011.83
64	0	-	-	3600.00	0	2675.63	2173.00	23.26	3600.00	0	3659.57	2066.00	77.41	3600.00
128	0	-	-	3600.00	0	8330.16	2309.00	261.52	3600.00	×	×	×	×	×
256	0	-	-	3600.00	0	19008.65	2481.00	666.65	3600.00	×	×	×	×	×
512	0	-	-	3600.00	0	45099.86	2512.00	1700.30	3600.00	×	×	×	×	×
1024	0	-	-	3600.00	0	116626.56	1573.00	7939.79	3600.00	×	×	×	×	×
2048	0	-	-	3600.00	0	592757.24	-	-	3600.00	×	×	×	×	×

in set *D250* with up to 64 links, as well as 13 instances with 128 links. Besides, it was able to find feasible solutions with average optimality gaps of 21.81%, 52.52%, and 192.72% for the instances in set *D250* with 256, 512, and 1024 links, respectively. However, B&C-F2 found optimal solutions for only the instances in sets *D10000* and *D250* with up to 64 links. This algorithm was not able to run on instances with 128 links or more. Besides, B&C-F1 was faster on average than B&C-F2 when both found optimal solutions.

Regarding the VRSP variant corresponding to the IEEE 802.11n standard, it can be seen from Tables 6 and 7 that Datarate PPTAS was able to find feasible solutions, within 3600 seconds, only for instances in set *D10000* with up to 32 links (as was the case in [6]), and for no instance in set *D250*. Meanwhile, B&C-F1 found optimal solutions for all instances in set *D10000* with up to 1024 links, and for all instances in set *D250* with up to 32 links. Besides, it was able to find feasible solutions for all instances in set *D10000* with 2048 links and for all instances in set *D250* with up to 1024 links, but with high optimality gaps. As above, B&C-F2 was able to run only on instances in both sets with up to 64 links. However, it found optimal solutions, within 3600 seconds, for all instances in set *D10000* with up to 64 links and all but one instance in set *D250* with 32 links.

Table 8
Results of B&C-F1 and B&C-F2 on set *D10000* regarding the VRBSP variant corresponding to the IEEE 802.11*ac* standard.

	B&C-F1					B&C-F2				
$\|L\|$	#opt	ub	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)
8	30	6240.00	6240.00	0.00	0.07	30	6240.00	6240.00	0.00	0.12
16	30	12480.00	12480.00	0.00	0.23	30	12480.00	12480.00	0.00	0.38
32	30	24960.00	24960.00	0.00	0.79	30	24960.00	24960.00	0.00	1.74
64	30	49909.60	49909.60	0.00	5.33	30	49909.60	49909.60	0.00	15.68
128	30	99788.00	99788.00	0.00	15.53	×	×	×	×	×
256	30	199392.70	199392.70	0.00	324.20	×	×	×	×	×
512	0	399250.47	254351.36	6198.23	3600.00	×	×	×	×	×
1024	×	×	×	×	×	×	×	×	×	×
2048	×	×	×	×	×	×	×	×	×	×

Table 9
Results of B&C-F1 and B&C-F2 on set *D250* regarding the VRBSP variant corresponding to the IEEE 802.11*ac* standard.

	B&C-F1					B&C-F2				
$\|L\|$	#opt	ub	lb	*gap*(%)	t(s)	#opt	ub	lb	*gap*(%)	t(s)
8	30	4750.53	4750.53	0.00	1.30	30	4750.53	4750.53	0.00	1.27
16	0	8810.10	7468.65	18.34	3600.00	0	8430.04	7391.21	14.30	3600.00
32	0	22126.66	10973.76	101.93	3600.00	0	21149.57	9313.46	127.60	3600.00
64	0	45993.53	13600.87	239.72	3600.00	0	42878.02	9033.91	388.07	3600.00
128	0	92330.71	8409.19	1775.77	3600.00	×	×	×	×	×
256	0	199680.00	–	–	3600.00	×	×	×	×	×
512	0	399360.00	–	–	3600.00	×	×	×	×	×
1024	×	×	×	×	×	×	×	×	×	×
2048	×	×	×	×	×	×	×	×	×	×

5.2. VRBSP

In this section, we report the experiments to evaluate the ability of B&C-F1 and B&C-F2 to obtain solutions for VRBSP. In this case the values for the minimum SINR q_m^b and its respective transmission rate $\bar{r}_m^b$, for all $b \in B = \{20, 40, 80, 160\}$ and $m \in M = \{0, \ldots, 9\}$, are defined accordingly to the IEEE standard 802.11*ac*. Therefore, the values of q_m^b and $\bar{r}_m^b$ are given by Table 1.

The results of B&C-F1 and B&C-F2 are reported in Tables 8 and 9, with Table 8 displaying the results for the instances in set *D10000* and Table 9 showing the results for the instances in set *D250*. These two tables have the same structure. Column 1 depicts the value of $|L|$, while the number of instances solved to optimality by B&C-F1 within 3600 seconds is depicted in Column 2. The average, over the 30 instances, of the upper bounds ub_i and lower bound lb_i, and the relative optimality gaps $(\frac{ub_i - lb_i}{lb_i})$ found by B&C-F1 are reported in columns 3 to 5, respectively. Besides, Column 6 displays the average running times of the same algorithm, respectively. The same results are presented for B&C-F2 in the last five columns, respectively. When one of these algorithms could not find a non-empty feasible solution within 3600 seconds, the corresponding column is filled with a dash '–'. Besides, when there was not enough memory to run any o these algorithms, the corresponding column is filled with an ×.

One can see from Tables 8 and 9 that B&C-F1 found optimal solutions for all instances in set *D10000* with up to 256 links, and for all instances in set *D250* with up to 8 links. Besides, it was able to find feasible solutions for all instances with 512 links in set *D10000* and for all instances in set *D250* with up to 128 links, but with high optimality gaps. Meanwhile, B&C-F2 was able to run only on instances in both sets with up to 64 links. It found optimal solutions, within 3600 seconds, for all instances with up to 64 links in set $D10000$ and all instances with 8 links in set $D250$. Besides, it was able to find feasible solutions for all instances in set *D250* with up to 64 links, but with high optimality gaps.

One can also see from these tables that the instances in set *D250* showed to be more challenging than those in set *D10000* of [6] as conjectured above. It can also be seen from Table 9 that on these harder instances, the upper bounds of B&C-F2 were better than those of B&C-F1. This was expected because the formulation F2 rely on variables w_{ij}^{uv}, for all $(i, j) \in L$ and $(u, v) \in L \backslash \{(i, j)\}$, instead of the Big M approach used in F1, which results in better linear relaxations upper bounds accordingly to [29,30]. Unfortunately, despite having better upper bounds than B&C-F1, B&C-F2 runs out of memory on smaller instances than B&C-F1.

6. Conclusion

In this paper, we dealt with the Variable Data Rate and Variable Bandwidth Scheduling Problem (VRBSP). In order to provide an algorithm to find optimal solutions for VRBSP and its special case VRSP, we proposed two MILP formulations, called F1 and F2. Computational experiments were carried out to evaluate the performance of the branch and cut algorithms based on F1 and F2, called B&B-F1 and B&C-F2, and to compare these algorithms with the best algorithm for VRSP in the literature, called Datarate PPTAS [6].

The results for VRSP showed that Datarate PPTAS is not computationally efficient, as it could only efficiently find feasible (non-optimal) solutions for instances in set $D10000$ with up to 64 links, and for none of the instances in set $D250$. B&C-F2 also showed performance issues. Due to the lack of RAM memory, it was able to run only on instances with up to 64 links. However, most of the solutions found for these instances were optimal. Best results were obtained by B&C-F1, which was able to find optimal solutions for all instances in set $D10000$ with up to 1024 links and 13 instances in set $D250$ with up to 128 links.

The results for VRBSP showed that the best results were again obtained by B&C-F1, which was able to find optimal solutions for all instances in set $D10000$ with up to 256 links, while B&C-F2 was able to run only on those instances in this set with up to 64 links. Regarding the instances in set $D250$, both branch and cut algorithms were able to find optimal solutions only for the smallest instances with 8 links. The upper bounds obtained by B&C-F2, within 3600 seconds were better than those of B&C-F1 on the instances in this set with 16, 32, and 64 links. However, the solutions (lower bounds) obtained by B&C-F1 within this time limit were better than those of B&C-F2.

Future works may study other approaches to linearize the mixed integer nonlinear formulation (4)–(10), based for example on the works of [31,32]. Besides, lifting approaches [33] can be applied to improve the upper bounds of F1, and Column Generation algorithms [34] can be used to efficiently compute the linear relaxation F2 on larger instances. Moreover, heuristics and metaheuristics [35] can be studied in order to solve the larger instances of VRBSP that cannot be tackled by the exact and approximation algorithms. The solutions obtained by these heuristics could also be used as a warm start for the MILP-based exact algorithms, as suggested in [36].

Declaration of Competing Interest

We declare that we have not competing interest regarding this paper.

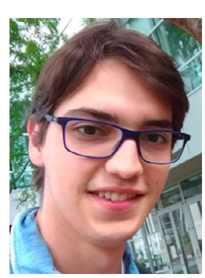
Acknowledgement

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, the Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brazil (CNPq), and the Fundação de Amparo á Pesquisa do Estado de Minas Gerais - Brazil (FAPEMIG).

References

[1] Cisco, 802.1 1ac: The Fifth Generation of Wi-Fi. Technical White Paper, Cisco and its affiliates, 2012.
[2] IEEE Computer Society, Standard for information technology - telecommunications and information exchange between systems local and metropolitan area networks - specific requirements - Part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications - amendment 4: enhancements for very high throughput for operation in bands below 6 GHz, Technical Report, 2013.
[3] O. Goussevskaia, L.F.M. Vieira, M.A.M. Vieira, Wireless multi-rate scheduling: from physical interference to disk graphs, in: Conference on Local Computer Networks, IEEE, 2012, pp. 651–658.
[4] E.H. Ong, J. Kneckt, O. Alanen, Z. Chang, T. Huovinen, T. Nihtilä, IEEE 802.11ac: enhancements for very high throughput wlans, in: International Symposium on Personal, Indoor and Mobile Radio Communications, IEEE, 2011, pp. 849–853.
[5] T. Kesselheim, Approximation algorithms for wireless link scheduling with flexible data rates, in: European Symposium on Algorithms, Springer, 2012, pp. 659–670.
[6] O. Goussevskaia, L.F. Vieira, M.A. Vieira, Wireless scheduling with multiple data rates: from physical interference to disk graphs, Comput. Netw. 106 (2016) 64–76.
[7] M.S. Gast, 802.11 Ac: a survival guide: Wi-Fi at gigabit and beyond, O'Reilly Media, Inc., 2013.
[8] P. Gupta, P.R. Kumar, The capacity of wireless networks, IEEE Trans. Inf. Theo. 46(2) (2000) 388–404.
[9] O. Goussevskaia, M.M. Halldórsson, R. Wattenhofer, E. Welzl, Capacity of arbitrary wireless networks, in: International Conference on Computer Communications (INFOCOM), IEEE, 2009, pp. 1872–1880.
[10] Y.-Y. Chen, C. Chen, Simulated annealing for interface-constrained channel assignment in wireless mesh networks, Ad Hoc Netw. 29 (2015) 32–44.
[11] T.S. Rappaport, Wireless communications: principles and practice, Prentice Hall, 2nd, 2002.
[12] M.M. Halldórsson, P. Mitra, Wireless capacity and admission control in cognitive radio, in: International Conference on Computer Communications (INFOCOM), IEEE, 2012, pp. 855–863.
[13] F.R. Vieira, J.F. de Rezende, V.C. Barbosa, Scheduling wireless links by vertex multicoloring in the physical interference model, Comput. Netw. 99 (2016) 125–133.
[14] A. Ephremides, T.V. Truong, Scheduling broadcasts in multihop radio networks, IEEE Trans. Commun. 38 (1990) 456–460.
[15] O. Goussevskaia, Y.A. Oswald, R. Wattenhofer, Complexity in geometric sinr, in: International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), ACM, 2007, pp. 100–109.
[16] G. Brar, D.M. Blough, P. Santi, Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks, in: International Conference on Mobile Computing and Networking (Mobicom), ACM, 2006, pp. 2–13.
[17] G.F. Riley, The georgia tech network simulator, in: Proceedings of the ACM SIGCOMM Workshop on Models, Methods and Tools for Reproducible Network Research, ACM, 2003, pp. 5–12.
[18] P. Djukic, S. Valaee, Delay aware link scheduling for multi-hop tdma wireless networks, IEEE/ACM Transactions on Networking (TON) 17 (2009) 870–883.
[19] R. Bellman, On a routing problem, Q. Appl. Math. 16 (1958) 87–90.
[20] L.R. Ford Jr., Network flow theory, Technical Report, The Rand Corporation, 1956.
[21] S. Chilukuri, A. Sahoo, Delay-aware tdma scheduling for multi-hop wireless networks, in: Proceedings of the 2015 International Conference on Distributed Computing and Networking, ACM, 2015. 27:1–27:10
[22] M. Andrews, M. Dinitz, Maximizing capacity in arbitrary wireless networks in the sinr model: Complexity and game theory, in: International Conference on Computer Communications (INFOCOM), IEEE, 2009, pp. 1332–1340.
[23] M.M. Halldórsson, P. Mitra, Wireless capacity with oblivious power in general metrics, in: Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, 2011, pp. 1538–1548.
[24] T. Kesselheim, A constant-factor approximation for wireless capacity maximization with power control in the sinr model, in: Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, 2011, pp. 1549–1559.
[25] S. Arora, D. Karger, M. Karpinski, Polynomial time approximation schemes for dense instances of np-hard problems, J. Comput. Syst. Sci. 58 (1999) 193–210.
[26] E.I. Ásgeirsson, M.M. Halldórsson, T. Tonoyan, Universal framework for wireless scheduling problems, in: International Colloquium on Automata, Languages, and Programming (ICALP), in: Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017. 129:1–129:15
[27] F. Glover, Improved linear integer programming formulations of nonlinear integer problems, Manage. Sci. 22 (4) (1975) 455–460.
[28] F. Glover, E. Woolsey, Converting the 0–1 polynomial programming problem to a 0–1 linear program, Oper. Res. 22 (1) (1974) 180–182.
[29] M.S. Bazaraa, J.J. Jarvis, H.D. Sherali, Linear programming and network flows, John Wiley & Sons, 2011.
[30] G. Codato, M. Fischetti, Combinatorial benders' cuts for mixed-integer linear programming, Oper. Res. 54 (2006) 756–766.
[31] A.R. Fügenschuh, E.M. Craparo, M. Karatas, S.E. Buttrey, Solving multistatic sonar location problems with mixed-integer programming, Optimiz. Eng. (2019) 1–31.
[32] F. Furini, E. Traversi, Theoretical and computational study of several linearisation techniques for binary quadratic problems, Ann. Oper. Res. 279 (1–2) (2019) 387–411.
[33] E. Balas, Projection, lifting and extended formulation in integer and combinatorial optimization, Ann. Oper. Res. 140 (1) (2005) 125.
[34] G. Desaulniers, J. Desrosiers, M.M. Solomon, Column generation, 5, Springer Science & Business Media, 2006.
[35] E.-G. Talbi, Metaheuristics: from design to implementation, Wiley Publishing, 2009.
[36] M. Fischetti, F. Glover, A. Lodi, The feasibility pump, Math. Program. 104 (1) (2005) 91–104.

José M. Costa has a Ph. D. from Universidade Federal de Minas Gerais (UFMG). He received his M. S. degree in Modelling Mathematics and Computational from Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG). His research interests are in Network Optimization and Computer Networking.



Pedro P. Paniago works as a Software Engineer at Google Brasil. He received his B.S. degree in Computer Mathematics from Universidade Federal de Minas Gerais (UFMG). His research interests are in Mathematical Programming and Computer Networking.

Joaquim de Andrade is a MSc Student at the Universidade Federal de Minas Gerais (UFMG), and have received his B.S degree at the Universidade Federal de Sergipe (UFS). His research interests are in Mathematical Programming, Combinatorial Optimization, and Metaheuristics.

Thiago F. Noronha is an Associate Professor of Computer Science at the Universidade Federal de Minas Gerais (UFMG). He received M.S. and Ph. D. degrees in Computer Science from Universidade Católica do Rio de Janeiro. His research interest is in Network Optimization.

Marcos A. M. Vieira is an Associate Professor of Computer Science at the Universidade Federal de Minas Gerais (UFMG) since 2011. He received M.S. and Ph. D. degrees in Computer Science from the University of Southern California (USC). His research interests are in Computer Networking, Wireless Networks, and Cyber-Physical Systems.