A lagrangian approach to chance constrained routing with local broadcast

Published Version:
Cacciola, M., Frangioni, A., Galli, L., Stea, G. (2021). A lagrangian approach to chance constrained routing with local broadcast. Cham : Springer Nature [10.1007/978-3-030-63072-0_22].

Availability:
This version is available at: https://hdl.handle.net/11585/983374 since: 2026-02-20

Published:
DOI: http://doi.org/10.1007/978-3-030-63072-0_22

A Lagrangian approach to Chance Constrained Routing with Local Broadcast

Matteo Cacciola, Antonio Frangioni, Laura Galli, and Giovanni Stea

Abstract Mobile cellular networks play a pivotal role in emerging Internet of Things (IoT) applications, such as vehicular collision alerts, malfunctioning alerts in Industry-4.0 manufacturing plants, periodic distribution of coordination information for swarming robots or platooning vehicles, etc. All these applications are characterized by the need of routing messages within a given local area (geographic proximity) with constraints about both timeliness and reliability (i.e., probability of reception). This paper presents a Non-Convex Mixed-Integer Nonlinear Programming model for a routing problem with probabilistic constraints on a wireless network. We propose an exact approach consisting of a branch-and-bound framework based on a novel Lagrangian decomposition to derive lower bounds. Preliminary experimental results indicate that the proposed algorithm is competitive with state-of-the-art general-purpose solvers, and can provide better solutions than existing highly tailored ad-hoc heuristics to this problem.

Key words: chance-constrained optimization, mixed-integer nonlinear programming, internet-of-things, mobile network routing, local broadcast, lagrangian relaxation, bundle methods, branch-and-bound.

1 Introduction

Long Term Evolution Advanced (LTE-A) technology for cellular networks is the new forefront in the context of transmission networks for location-based broadcast

A. Frangioni, L. Galli, M. Cacciola
Dipartimento di Informatica, Università di Pisa, Largo Bruno Pontecorvo 3, 56127 Pisa, Italy, e-mail: {antonio.frangioni, laura.galli}@unipi.it, m.cacciola1@studenti.unipi.it

G. Stea
Dipartimento di Ingegneria dell'Informazione, Università di Pisa, Largo Lucio Lazzarino 1, 56122, Pisa, Italy e-mail: {giovanni.stea}@unipi.it

services, such as advertising, smart-city applications, and Internet-of-Things (IoT) deployments. Yet, some new IoT services, such as vehicular collision alerts and augmented-reality live games, require *low latency* and *high reliability*, as well as the possibility to target an area defined by the application itself rather than the cell coverage. While traditional LTE-A tools can support these services, they do so at a rather large cost in terms of energy. In fact, on the one hand, LTE's built-in Multicast/Broadcast mechanism was originally devised for broadcasting multimedia, and therefore unsuitable to this task because it is static: the message transmission format, the target area and the period of broadcast transmissions must all be selected statically. On the other hand, having the base station (antenna), called eNodeB (eNB) in the LTE terminology, relay messages to all the User Equipment (UEs) in a target area using unicast downlink (DL) transmissions (one per targeted UE) would require too many DL resources, hence too much energy. For this reason, recently, a new communication framework has been proposed. We consider a network of mobiles (UEs) which are under the control of a single eNB, as shown in Figure 1.

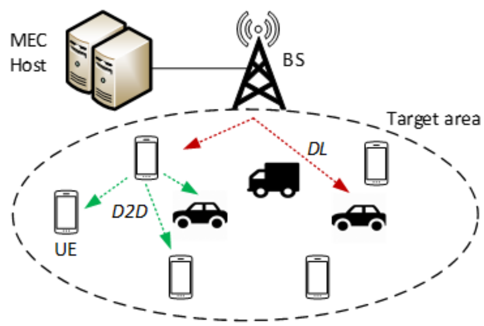

Fig. 1 System model

The eNB can send them information using DL (i.e., vertical) transmissions. Information can also travel through device-to-device (D2D) links (i.e. horizontal broadcast transmissions originated at UEs). Vertical links are reliable but costly, and should be avoided if possible. By contrast, horizontal transmissions are free (from the eNB viewpoint), but not reliable: there is no ARQ mechanism involved, and it is impractical to try and ascertain which UEs, in the neighbourhood of the transmitter, have successfully decoded a message. However, UEs can act as multi-hop relays: horizontal transmissions are scheduled by the eNB, which issues grants to the UEs that may transmit. This allows to model the probability that a certain horizontal transmission is successful with a reasonable accuracy, given the position of the UEs, the transmission power of the transmitter, and the modulation and coding scheme adopted for transmission. This yields a new *Chance-Constrained Unicast-Multicast Routing Problem* (CCUMRP): select which vertical and horizontal multi-hop transmissions to choose in order to guarantee that all UEs receive the information with a certain level of reliability within a certain time limit, at minimum energy cost. We propose a Non-Convex Mixed-Integer Nonlinear Programming (MINLP) model for CCUMRP, together with an ad-hoc Lagrangian decomposition approach to compute lower bounds that separates the variable of the problem in a somewhat unusual fashion. We use the latter as the basis of a Branch-and-Bound (B&B) approach that we computationally compare both with state-of-the-art, general-purpose exact solvers and with highly tailored ad-hoc heuristics for the problem.

We model the *system* as a graph $G = (N, A)$, where $N = \{0\} \cup N'$ (0 being the eNB and N' representing the UEs) with $n = |N'|$, while the arc set $A = A' \cup A''$ consists of two types of arcs:

- *vertical* arcs A' of the form $(0, i)$ for all $i \in N'$, representing a DL transmission between the eNB the UE i having probability 1 to be decoded successfully at i but high energy cost;
- *horizontal* arcs A'' of the form (i, j) for $i \neq j \in N'$, representing a D2D transmission from i to j having probability $0 < P_{ij} < 1$ to be decoded successfully at j, but low (energy) cost. (Probabilities are independent.)

In the initial stage of the process the eNB transmits the message to a subset of UEs via DL (i.e., vertical transmission), while in the following stages only horizontal transmissions are allowed. A node $i \in N'$ can only issue an horizontal transmission at a given stage if granted permission from the eNB. At most M grants can be assigned in each stage, to ensure that the ensuing transmissions are not mutually interfering. The problem is therefore to transmit the message to the entire floorplan with the following constraints:

1. To ensure timeliness of reception of the message to all UEs, the broadcast process must be over in k stages, with k known a priori. Because the first round is clearly "special" (vertical transmissions from eNB to UEs), it is useful to define the set $K' = \{2, \ldots, k\}$ of "normal" stages (horizontal transimissions between UEs).
2. To ensure reliability of reception, at the end of the broadcast process, each UE must possess the message with at least a given probability α.

The main *objective* is to reduce the number of vertical transmissions. However, besides them, we also need to decide which UEs should transmit when (i.e., in which stage), so we must define the schedule of the grants that the eNB has to issue in order to compose the broadcast schedule. A secondary objective is to issue the least possible numbers of grants. In our model it is actually easy to generalize this by considering node-and-stage weighted grants costs β_i^h ($i \in N'$, $h \in K'$), e.g., depending on the type of node i and/or its remaining battery power.

2 Mathematical Model and Decomposition

We define the following set of *variables*:

- binary x_i for $i \in N'$ indicating whether node i is selected in the initial set of UEs that are reached by the vertical transmission from eNB at stage 1;

- continuous $p_i^h \in [\,0\,,\,1\,]$ for $i \in N'$ and $h \in K$ indicating the total probability that node i has been reached at all stages up to h;
- binary g_i^h for $i \in N'$ and $h \in K'$ indicating whether node i is selected to receive a grant for broadcasting at stage h.

The MINLP *model* of CCUMRP is as follows:

$$\min \textstyle\sum_{i \in N'} x_i + \sum_{h \in K'} \sum_{i \in N'} \beta_i^h g_i^h \tag{1}$$

$$p_i^1 = x_i \qquad i \in N' \tag{2}$$

$$p_i^k \geq \alpha \qquad i \in N' \tag{3}$$

$$1 - p_i^h \geq (\,1 - p_i^{h-1}\,) \textstyle\prod_{(j,i) \in A''} (\,1 - g_j^h p_j^{h-1} P_{ji}\,) \qquad i \in N' \;,\; h \in K' \tag{4}$$

$$\textstyle\sum_{i \in N'} g_i^h \leq M \qquad h \in K' \tag{5}$$

$$x_i \in \{\,0\,,\,1\,\} \qquad i \in N' \tag{6}$$

$$0 \leq p_i^h \leq 1 \qquad i \in N' \;,\; h \in K \tag{7}$$

$$g_i^h \in \{\,0\,,\,1\,\} \qquad i \in N' \;,\; h \in K' \tag{8}$$

The objective function (1) minimizes the number of initial vertical transmissions used in the first stage ($h = 1$) and the cost of grants issued during the subsequent stages ($h \in K'$); it is therefore intended that $\beta_i^h \ll 1$. Constraints (2) ensure that all initially targeted nodes are certainly reached. Constraints (3) impose that each UE node $i \in N'$ is ultimately (at stage k) reached with probability at least α; clearly, it would be trivial to generalize the model by allowing node-specific thresholds α_i. Constraints (5) bound the total number of grants available at each stage (again, it would be trivial to let M depend on h). Finally, the constrains characterizing the model are the *nonlinear nonconvex* (4) ones, expressing the probability that node i at stage h has *not* yet received the message.

Clearly, the problem would be almost trivial were it not for (4); therefore, it is those we will concentrate upon. Taking logarithms and noting that $g_j^h = 0 \implies \log(\,1 - g_j^h p_j^{h-1} P_{ji}\,) = 0$ they can be reduced to

$$\log(1 - p_i^h) \geq \log(\,1 - p_i^{h-1}\,) + \textstyle\sum_{(j,i) \in A''} g_j^h \log(\,1 - p_j^{h-1} P_{ji}\,) \tag{9}$$

which is at least linear with respect to variables g_j^h. However, the logarithm is ill-defined when $p_i^h = 1$, which certainly happens at least whenever $x_i = 1$. We therefore consider a *restriction* of the problem by selecting a constant $\bar{p} < 1$ "arbitrarily close to 1", replacing (2) and (7), respectively, with

$$p_i^1 = x_i \bar{p} \qquad i \in N' \tag{10}$$

$$0 \leq p_i^h \leq \bar{p} \qquad i \in N' \;,\; h \in K' \tag{11}$$

Clearly, each feasible solution of the new model is feasible for the original one as well, and by choosing $\bar{p}$ appropriately the practical difference between the two is poised to be minimal. Finally, let us mention for future reference that for the second stage (i.e., $h = 2$) constraints (9) can be written in the form

$$\log(1-p_i^2) \geq \log(1-\bar{p})x_i + \textstyle\sum_{(j,i)\in A''} g_j^2 \log(1-P_{ji})x_j \qquad i \in N' \ , \qquad (12)$$

whose useful property is that the right-hand side does not contain any continuous variable (the p_i^1 having been substituted with the x_i). Therefore, (4) can be replaced by (9) for $h \in K' \setminus \{2\}$ and by (12) for $h = 2$. Nor that this, by itself, makes the constraints significantly easier to deal with. However, it allows us to propose a decomposition approach to compute globally valid lower bounds. In particular, we present a *Lagrangian decomposition* of the MINLP formulation

$$\min \textstyle\sum_{i\in N'} x_i + \sum_{h\in K'} \sum_{i\in N'} \beta_i^h g_i^h$$
$$(10)\ ,\ (3)\ ,\ (12)\ ,\ (9)\ ,\ (5)\ ,\ (6)\ ,\ (11)\ ,\ (8)$$

The idea is to relax constraints (12) and (9) with *Lagrangian multipliers* $\lambda_i^h \geq 0$ for $i \in N'$ and $h \in K'$. In so doing, the problem is decomposed into k separate sub-problem; this is clearly due to the fact that (12)/(9) are the only constraints that link the variables of one stage to those of the following one. One may expect that each sub-problem has the variables corresponding to one specific level $h \in K$, but in fact the decomposition is somewhat different, and perhaps somewhat unusual. Indeed, each sub-problem actually has variables "of one kind" for one stage h and variables "of another kind" for the subsequent stage $h+1$ (if any). This is due to the terms $g_j^h \log(1-p_j^{h-1}P_{ji})$ in (9) (and, similarly, $g_j^2 \log(1-P_{ji})x_j$ in (12)) that link together variables g_j^h with variables p_j^{h-1} (x_j). We will now describe the sub-problems. Due to the special nature of the first stage, the corresponding sub-problem clearly has a particular structure. Not having a subsequent stage, the sub-problem corresponding to the last stage also has a peculiar form. All the sub-problems corresponding to intermediate stages rather share the same structure.

The *first sub-problem* ($h = 1$) contains the x_i variables (that substitute for the probability variables p_i^1) of the first stage and the grant variables g_i^2 of the second stage, reading

$$\begin{aligned} \min\ & \textstyle\sum_{i\in N'} \big[\, x_i + \beta_i^2 g_i^2 + \lambda_i^2 \big(\log(1-\bar{p})x_i + \sum_{(j,i)\in A''} g_j^2 \log(1-P_{ji})x_j \big) \big] && (13) \\ & \textstyle\sum_{i\in N'} g_i^2 \leq M \\ & x_i \ ,\ g_i^2 \in \{0,1\} && i \in N' \end{aligned}$$

Collecting like terms of (13), and observing that there is no point in setting $g_i^2 = 1$ if $x_i = 0$, yields:

$$\begin{aligned} \min\ & \textstyle\sum_{i\in N'} \big[\, (1+\lambda_i^2 \log(1-\bar{p}))x_i + \big(\beta_i^2 + \sum_{(i,j)\in A''} \lambda_j^2 \log(1-P_{ij})\big) g_i^2 \,\big] \\ & \textstyle\sum_{i\in N'} g_i^2 \leq M \\ & g_i^2 \leq x_i && i \in N' \\ & x_i \ ,\ g_i^2 \in \{0,1\} && i \in N' \end{aligned}$$

Since all nonlinear operations are applied to constants, the problem is linear. Furthermore, the special structure of the constraints ensures that, despite the variables being integer-valued, it can easily be solved in $\mathscr{O}(n \log n)$.

Next, each of *sub-problems* ($2 < h < k$) contains grant variables g_i^{h+1} of stage $h+1$ and probability variables p_i^h of stage h, reading

$$\begin{aligned} \min \textstyle\sum_{i \in N'} \big[& (\lambda_i^{h+1} - \lambda_i^h) \log(1 - p_i^h) + \\ & \big(\beta_i^{h+1} + \textstyle\sum_{(i,j) \in A''} \lambda_j^{h+1} \log(1 - p_i^h P_{ij}) \big) g_i^{h+1} \big] \qquad (14) \\ & 0 \le p_i^h \le \bar{p} \qquad i \in N' \\ & \textstyle\sum_{i \in N'} g_i^{h+1} \le M \qquad (15) \\ & g_i^{h+1} \in \{ 0 , 1 \} \qquad i \in N' \qquad (16) \end{aligned}$$

Clearly, in this problem each variable p_i^h only interacts with the others via the single term in which it is multiplied by the corresponding g_i^{h+1}. The term is highly nonlinear, but still one can consider the corresponding function

$$f_i^h(p , g) = (\lambda_i^{h+1} - \lambda_i^h) \log(1 - p) + \big(\beta_i^{h+1} + \textstyle\sum_{(i,j) \in A''} \lambda_j^{h+1} \log(1 - pP_{ij}) \big) g \ .$$

By computing the two costants $p_i^{h,g} = \operatorname{argmin}\{ f_i^h(p , g) \ : \ 0 \le p \le \bar{p} \}$ for $g \in \{ 0 , 1 \}$, the sub-problem can be rewritten as

$$\min \big\{ \textstyle\sum_{i \in N'} f_i^h(p_i^{h,1} , 1) g_i^{h+1} + f_i^h(p_i^{h,0} , 0)(1 - g_i^{h+1}) \ : \ (15) \ , \ (16) \big\}$$

and, therefore, again easily solved in $\mathscr{O}(n \log n)$. The crux of the subproblem therefore lies in the computation of $p_i^{h,1}$ and $p_i^{h,0}$. Computing the latter is trivial, as it reduces to minimizing on $p \in [0 , \bar{p}]$ the monotone function $(\lambda_i^{h+1} - \lambda_i^h) \log(1 - p)$; the optimum necessarily lies in one of the two extremes. Finding $p_i^{h,1}$, instead, requires to tackle a more complex *one-dimensional minimization problem* of the form

$$\min \big\{ f(p) = c \log(1 - p) + \textstyle\sum_{i \in N'} a_i \log(1 - pb_i) \ : \ 0 \le p \le \bar{p} \big\} \qquad (17)$$

where, $a_i = \lambda_i^{h+1} \ge 0$, $0 \le P_{ji} = b_i < 1$, $c = \lambda_i^{h+1} - \lambda_i^h$ is unrestricted in sign, and whose solution is discussed below.

Finally, the remaining *Lagrangian term for k* is

$$\min \big\{ \textstyle\sum_{i \in N'} -\lambda_i^k \log(1 - p_i^k) \ : \ \alpha \le p_i^k \le \bar{p} \big\}$$

that is separable over i; being the objective convex ($\lambda_i^k \ge 0$), the optimum is in the left endpoint $p_i^k = \alpha$.

The crucial step is clearly the ability to efficiently solve the *one-dimensional problem* (17). Yet, if $c \ge 0$ then the problem is trivial: $f(p)$ is a decreasing function with $\lim_{p \to 1^-} f(p) = -\infty$, so the minimum is attained at $\bar{p}$. We will therefore concentrate on the case where $c < 0$ instead, for which we will prove that there is *at most one* critical point $p_0 \in [0 , \bar{p}]$; moreover, the minimum is either attained at 0 or p_0.

Indeed, $f(p)$ is the sum of the increasing function $c\log(1-p)$ ($c<0$) with vertical asymptote at $p=1$, and n decreasing functions $a_i \log(1-pb_i)$ ($a_i \geq 0$) with vertical asymptotes at $p = 1/b_i > 1$ (since $b_i < 1$). Hence, clearly as $p \to 1$ the increasing function dominates: $\lim_{p\to 1^-} f(p) = +\infty$, and $f(p)$ has to be strictly increasing "close to" $\bar{p}$. As p approaches 0, instead, the behaviour depends on the a_i values. In particular, we prove the following two cases: *either* the function is decreasing in 0 and becomes increasing "closer to" $\bar{p}$, which implies that the minimum is attained in the interior, *or* the function is increasing in 0 and remains so in the whole interval, which implies that the minimum is attained at 0.

Lemma 1. *If $c < 0$, $0 \leq b_i \leq 1$ and $a_i \geq 0$ then there exists* at most one *critical point $p_0 \in [0,1)$ such that $f'(p_0) = 0$, and $f(p)$ is* strictly increasing *in $p_0 < p \leq 1$.*

Proof. Consider $f'(p) = -\frac{c}{1-p} - \sum_{i\in N'} \frac{a_i b_i}{1-pb_i}$, we have

$$f'(p) \geq 0 \quad \Longleftrightarrow \quad -\sum_{i\in N'} \frac{(1-p)a_i b_i}{(1-pb_i)c} = h(p) \leq 1 \ .$$

It is now immediate to see that

$$h'(p) = -\sum_{i\in N'} \frac{a_i b_i}{c} \frac{b_i - 1}{(1-pb_i)^2} \leq 0$$

for all $p \in [0,1]$. This means that there can be at most one point $p_0 \in [0,1)$ such that $f'(p_0) = 0$, and therefore $f(p)$ is *strictly increasing* in $(p_0, 1)$. □

We now analyse convexity of f, showing that if the function is non-convex then there is exactly one point $\hat{p}$ in which the second derivative changes its sign, and that the function is convex in $[\hat{p}, 1]$.

Lemma 2. *If $c < 0$, $0 \leq b_i \leq 1$ and $a_i \geq 0$, then there exists* at most one *point $\hat{p} \in [0,1)$ with $f''(\hat{p}) = 0$, and $f(p)$ is* convex *in $\hat{p} \leq p \leq 1$.*

Proof. Along the same lines, for $f''(p) = -\frac{c}{(1-p)^2} - \sum_{i\in N'} \frac{a_i b_i^2}{(1-pb_i)^2}$ we have

$$f''(p) \geq 0 \quad \Longleftrightarrow \quad -\sum_{i\in N'} \frac{(1-p)^2 a_i b_i^2}{(1-pb_i)^2 c} = h(p) \leq 1$$

which similarly yields

$$h'(p) = -\sum_{i\in N'} \frac{a_i b_i^2}{c} 2(1-p)(1-pb_i) \frac{b_i - 1}{(1-pb_i)^4} < 0$$

for all $p \in [0,1]$. Again, this implies that if $f''(\hat{p}) = 0$ for some $\hat{p} \in [0,1)$, then $f''(p) \geq 0$ (i.e., f is convex) for all $\hat{p} \leq p \leq 1$. □

To recap, the following *three* cases can happen:

1. $f(p)$ is increasing in $[0\,,\,\bar{p}]$, hence the minimum is 0;
2. $f(p)$ is decreasing in 0 but convex in $[0\,,\,\bar{p}]$, hence the minimum is in the interior of the interval;
3. $f(p)$ is decreasing in 0 and convex in $[\hat{p}\,,\,\bar{p}]$ for some $\hat{p} > 0$, hence the minimum lies in the interval $[\hat{p}\,,\,\bar{p}]$;

that are represented in Figures 2, 3, 4, respectively.

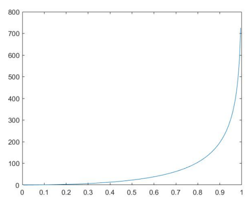

Fig. 2 Increasing

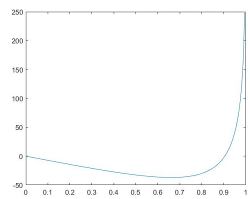

Fig. 3 Convex

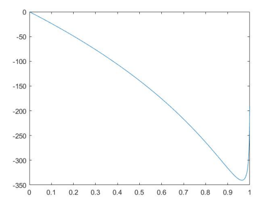

Fig. 4 Concave, then convex

From an algorithmic viewpoint, such a function can be efficiently globally minimized using a simple globalization of *Newton's method*. We keep an interval $[p_-\,,\,p_+]$ such that $f'(p_-) < 0$ and $f'(p_+) > 0$ (initialized as $[0\,,\,\bar{p}]$, unless $f'(0) \geq 0$ in which case we immediately terminate). If $f''(p_-) < 0$ (f is non convex at p_-) we use a simple bisection rule to find a point $p_- < p' < p_+$, we compute $f'(p')$ and shrink the interval accordingly. Otherwise (f is convex at p_-) we compute Newton's step, and we accept it if it belongs to the interval and it shrinks it enough; otherwise we revert to the simple bisection rule. This is clearly convergent, and typically quadratically so in the tail. Note that in our function the minimum is often close to 1, so instead of using a standard bisection we use the point $p' = p_- + 3/4(p_+ - p_-)$, as this typically leads to faster initial convergence.

3 Algorithmic Approaches and Experiments

Due to space restrictions, we briefly discuss the algorithmic approach that we developed using the proposed model and decomposition method.

It is well-known that for each choice of $\lambda \in \mathbb{R}^{2n}_{+}$, the solution of the corresponding *Lagrangian relaxation* provides a valid global lower bound on the optimal value of the original problem. To find the best possible Lagrangian relaxation, one then has to solve the *Lagrangian Dual* problem, i.e., maximize over all $\lambda \geq 0$ the *Lagrangian function* consisting of the sum of the k terms previously described. The efficiency of the solution process obviously depend on the specific algorithm used to solve the Lagrangian Dual; in our case we use the freely available implementation of the *(generalized) proximal Bundle method* [3] already used with success in other applications (e.g., [7, 6]) provided by the `NDOSolver/FiOracle` suite of `C++`

solvers for NonDifferentiable Optimization problems developed by the Department of Computer Science of the University of Pisa [9]. We refer to [3] and [9] for details on Bundle methods.

A fundamental component of any partial enumeration approach are the *heuristics* used to produce good feasible solutions that can be used to prune nodes of the decision tree (and that ultimately provide the returned best solution). We do this, potentially at each iteration, using both the integer, but (typically) not feasible, solution that we obtain by computing the Lagrangian function, and the continuous, but (quickly) “almost feasible”, convexified solution that can be obtained as a by-product of solving the master problem in the Bundle method [4]. Actually, exploiting both synergistically has been shown to be useful in some applications [2, 5].

Since both upper and lower bounds obtained with the methods previously discussed are not very tight, we implemented an *implicit enumeration* (Branch-and-Bound) algorithm in order to obtain better gaps.

We tested the model on the realistic scenarios constructed with the help of the `SimuLTE` simulator developed at the Department of Information Engineering of the University of Pisa [10]. The tool allows to create many instances of the problem tuning the main parameters of interests; in our experiments we mainly concentrated on the number of UEs, on the radius (in meters) of the geographical region of interest, and on the required coverage probability α.

We compared our Lagrangian-based B&B with the state-of-the-art, general-purpose MINLP solver BARON [1] 18.11.12, as well as with a highly-tailored combinatorial heuristic available in `SimuLTE` and described in [8]. For BARON, we scaled the objective function by a factor of 5 so that all the coefficients are integer, allowing it to also exploit integrality to round up the lower bound. All codes have been compiled with `g++` 7.4.0 and ran single-threaded on a machine sporting a 16-core Intel Xeon5120 CPU@2.20GHz and 64Gb RAM, running Ubuntu 18.04. The results are reported in the following Tables, with two different time limits: 300 seconds and 3000 seconds. The instances are characterized by the number of UEs (“#”), the radius (“*r*”) and the covering probability (“α”). For both exact methods we report the total running time (“time”) if they terminated before the time limit, and “–” otherwise, plus the total number of B&B nodes (“nodes”). We also report the *inherent gap* (“gap”), i.e., $(UB - LB)/\max\{1, LB\}$ (in percentage), where UB and LB are the best upper and lower bound on the optimal value produced by the corresponding algorithm at termination. To better represent the relative quality of the upper and lower bounds, we also separately report the *primal gap* (“pgap”) $(UB - \underline{UB})/\max\{1, \underline{UB}\}$ and the *dual gap* (“dgap”) $(\overline{LB} - LB)/\max\{1, LB\}$ (in percentage), where $\underline{UB}$ and $\overline{LB}$ are, respectively, the best (lowest) known upper bound and best (highest) known lower bound on the optimal value of the instance. Note that since the largest and hardest instances were not solved within 3000 seconds, a 0 primal or dual gap does necessarily means that the corresponding *UB*/*LB* are the optimal value, but only that they are the best ever found in our experiments.

The results in Table 1 clearly show how challenging CCUMRP is. Only 10-UEs instances can be all solved to optimality by our approach within the 5-minutes time limit; it is generally more efficient than BARON (which fails to solve two) except

Instances			BARON					B&B					CH
#	r	α	time	nodes	gap	pgap	dgap	time	nodes	gap	pgap	dgap	pgap
10	100	0.92	4.86	1	0.00	0.00	0.00	0.59	20	0.00	0.00	0.00	0.00
10	100	0.95	3.07	1	0.00	0.00	0.00	0.58	20	0.00	0.00	0.00	0.00
10	100	0.96	3.43	1	0.00	0.00	0.00	0.67	20	0.00	0.00	0.00	0.00
10	250	0.92	4.92	1	0.00	0.00	0.00	0.44	20	0.00	0.00	0.00	0.00
10	250	0.95	75.39	1	0.00	0.00	0.00	0.73	20	0.00	0.00	0.00	0.00
10	250	0.96	31.32	1	0.00	0.00	0.00	0.46	20	0.00	0.00	0.00	71.4
10	500	0.92	80.67	84	0.00	0.00	0.00	193.8	12323	0.00	0.00	0.00	71.4
10	500	0.95	44.45	52	0.00	0.00	0.00	44.00	2717	0.00	0.00	0.00	40.0
10	500	0.96	383.4	1597	0.00	0.00	0.00	229.1	5130	0.00	0.00	0.00	46.7
10	750	0.92	269.2	1778	0.00	0.00	0.00	153.42	2402	0.00	0.00	0.00	29.4
10	750	0.95	–	715	4.00	0.00	4.00	208.5	6880	0.00	0.00	0.00	38.5
10	750	0.96	–	717	13.0	0.00	13.0	29.87	1026	0.00	0.00	0.00	50.0
10	1000	0.92	1.78	1	0.00	0.00	0.00	79.81	2913	0.00	0.00	0.00	26.9
10	1000	0.95	1.42	1	0.00	0.00	0.00	210.0	13754	0.00	0.00	0.00	36.4
10	1000	0.96	0.82	1	0.00	0.00	0.00	1.91	120	0.00	0.00	0.00	63.6
25	100	0.92	–	1	3780	3050	120	121.6	164	0.00	0.00	0.00	0.00
25	100	0.95	–	17	100	0.00	100	98.45	130	0.00	0.00	0.00	0.00
25	100	0.96	–	18	80.0	0.00	80	57.56	84	0.00	0.00	0.00	0.00
25	250	0.92	–	1	3780	3050	120	12.83	58	0.00	0.00	0.00	0.00
25	250	0.95	–	1	3780	2600	140	11.30	58	0.00	0.00	0.00	0.00
25	250	0.96	–	1	3780	2600	140	10.04	56	0.00	0.00	0.00	0.00
25	500	0.92	–	1	3780	1354	260	–	1995	40.0	7.69	30.0	30.8
25	500	0.95	–	1	3780	1354	260	–	1983	23.1	23.1	0.00	69.2
25	500	0.96	–	1	3780	1250	260	–	1569	23.1	14.3	0.00	42.9
25	750	0.92	–	2	1160	456	80	–	642	40.0	2.94	8.00	32.4
25	750	0.95	–	5	1081	425	88	–	1263	23.3	2.78	0.00	22.2
25	750	0.96	–	4	1081	425	88	–	1004	23.3	2.78	0.00	22.2
25	1000	0.92	–	12	330	210	25	–	1332	14.6	3.28	0.00	36.2
25	1000	0.95	–	10	311	205	26	–	1185	12.5	1.61	3.57	29.0
25	1000	0.96	–	12	294	200	25	–	951	12.3	1.59	5.26	47.6
50	100	0.92	–	1	6280	5133	40	–	283	100	0.00	20.0	0.00
50	100	0.95	–	1	6280	5133	40	–	283	100	0.00	20.0	0.00
50	100	0.96	–	1	6280	5133	40	–	283	100	0.00	20.0	0.00
50	250	0.92	–	1	780	550	80	–	283	60.0	0.00	20.0	0.00
50	250	0.95	–	1	6280	4386	80	–	283	80.0	0.00	20.0	0.00
50	250	0.96	–	1	6280	4386	80	–	284	80.0	0.00	20.0	0.00
50	500	0.92	–	1	6280	2143	140	–	283	180	0.00	40.0	21.4
50	500	0.95	–	1	6280	1993	160	–	284	114	0.00	14.3	20.0
50	500	0.96	–	1	6280	1993	160	–	283	87.5	0.00	0.00	20.0
50	750	0.92	–	1	6280	913	200	–	292	230	6.45	0.00	3.20
50	750	0.95	–	1	6280	772	260	–	291	192	5.56	0.00	22.2
50	750	0.96	–	1	6280	749	260	–	290	185	0.00	0.00	18.9
50	1000	0.92	–	1	6280	398	560	–	283	220	1.59	40.0	11.1
50	1000	0.95	–	1	6280	376	600	–	280	156	4.55	11.1	28.8
50	1000	0.96	–	1	6280	355	700	–	280	154	2.90	25.0	31.9

Table 1 Computational results, time limit 300 seconds

for very large r, where BARON closes at root node. Interestingly, the combinatorial heuristic—which is the state-of-the-art for the problem up until this work—provides solutions that can be in excess of 50% off the optimum, although of course does so in orders-of-magnitude less time. When the size of the instances grows, BARON is basically unable to solve the problem except in a handful of cases, providing both lower and especially upper bounds that are of no practical value. Our approach cannot be exactly deemed to be very successful, with final gaps up to 40% with 25 users and even in excess of 200% with 50 users; however, it still produces the best solutions and lower bounds.

Moving to the time limit of 3000 seconds, depicted in Table 2, confirms that our approach at least scales much better than BARON; the much faster bound computation allows to enumerate more B&B nodes, which ultimately results in much better upper and lower bounds. In particular, we are able to solve about half of the instances with 25 users to optimality, with the other half ending with "reasonable" gaps (at least, if compared with these of BARON, both lower and upper, and with the upper bounds provided by the combinatorial heuristic). All in all, our approach is only partly successful. In particular, the lower bound is not particularly tight, which limits the size of the instances that can be practically solved. However, it at least provides a way to assess the performances of the heuristics approaches which, due to the extremely tight time limits (a handful of milliseconds) imposed by the application, are probably the only practical way of approaching the problem. Hopefully, the information provided by our approach will allow to better identify the limits of the current heuristics, and develop better ones.

Acknowledgements The authors gratefully acknowledge the partial financial support from the Italian Ministry of Education, University and Research (MIUR), under the project "Nonlinear and combinatorial aspects of complex networks" (grant `PRIN 2015B5F27W`).

References

1. The BARON solver, https://minlp.com/baron.
2. A. Borghetti, A. Frangioni, F. Lacalandra, and C.A. Nucci. Lagrangian Heuristics Based on Disaggregated Bundle Methods for Hydrothermal Unit Commitment. *IEEE Transactions on Power Systems*, 18(1):313–323, February 2003.
3. A. Frangioni. Generalized Bundle Methods. *SIAM J. on Optimization*, 13(1):117–156, 2002.
4. A. Frangioni. About Lagrangian Methods in Integer Optimization. *Annals of Operations Research*, 139(1):163–193, 2005.
5. A. Frangioni, C. Gentile, and F. Lacalandra. Solving Unit Commitment Problems with General Ramp Contraints. *International J. of Electrical Power and Energy Systems*, 30:316–326, 2008.
6. A. Frangioni and E. Gorgone. Generalized Bundle Methods for Sum-Functions with "Easy" Components: Applications to Multicommodity Network Design. *Mathematical Programming*, 145(1):133–161, 2014.
7. A. Frangioni, A. Lodi, and G. Rinaldi. New approaches for optimizing over the semimetric polytope. *Mathematical Programming*, 104(2-3):375–388, 2005.
8. G. Nardini, G. Stea, and A. Virdis. Supporting critical alert services through scheduled device-to-device transmissions in cellular networks. working paper (submitted), 2019.

Instances			BARON					B&B					CH
#	r	α	time	nodes	gap	pgap	dgap	time	nodes	gap	pgap	dgap	pgap
10	100	0.92	5.13	1	0.00	0.00	0.00	0.59	20	0.00	0.00	0.00	0.00
10	100	0.95	3.12	1	0.00	0.00	0.00	0.58	20	0.00	0.00	0.00	0.00
10	100	0.96	3.48	1	0.00	0.00	0.00	0.67	20	0.00	0.00	0.00	0.00
10	250	0.92	4.69	1	0.00	0.00	0.00	0.44	20	0.00	0.00	0.00	0.00
10	250	0.95	75.22	1	0.00	0.00	0.00	0.73	20	0.00	0.00	0.00	0.00
10	250	0.96	31.12	1	0.00	0.00	0.00	0.46	20	0.00	0.00	0.00	71.4
10	500	0.92	82.39	84	0.00	0.00	0.00	193.7	12323	0.00	0.00	0.00	71.4
10	500	0.95	46.08	52	0.00	0.00	0.00	44.0	2717	0.00	0.00	0.00	40.0
10	500	0.96	383.4	1597	0.00	0.00	0.00	229.1	5130	0.00	0.00	0.00	46.7
10	750	0.92	269.2	1778	0.00	0.00	0.00	153.42	2402	0.00	0.00	0.00	29.4
10	750	0.95	439.0	911	0.00	0.00	0.00	208.5	6880	0.00	0.00	0.00	38.5
10	750	0.96	1456	2605	0.00	0.00	0.00	29.9	1026	0.00	0.00	0.00	50.0
10	1000	0.92	1.78	1	0.00	0.00	0.00	79.81	2913	0.00	0.00	0.00	26.9
10	1000	0.95	1.66	1	0.00	0.00	0.00	210.0	13754	0.00	0.00	0.00	36.4
10	1000	0.96	0.90	1	0.00	0.00	0.00	1.91	120	0.00	0.00	0.00	63.6
25	100	0.92	–	71	100	0.00	100	121.6	164	0.00	0.00	0.00	0.00
25	100	0.95	–	94	80.0	0.00	80.0	98.5	130	0.00	0.00	0.00	0.00
25	100	0.96	–	109	80.0	0.00	80.0	57.6	84	0.00	0.00	0.00	0.00
25	250	0.92	–	107	80.0	0.00	80.0	12.8	58	0.00	0.00	0.00	0.00
25	250	0.95	–	21	100	0.00	100	11.3	58	0.00	0.00	0.00	0.00
25	250	0.96	–	18	100	0.00	100	10.0	56	0.00	0.00	0.00	0.00
25	500	0.92	–	29	3680	1354	160	–	5300	27.3	7.69	18.2	30.8
25	500	0.95	–	28	3050	1354	117	–	7552	7.69	7.69	0.00	69.2
25	500	0.96	–	39	3050	1250	117	–	6404	23.1	14.3	0.00	42.9
25	750	0.92	–	35	950	456	50.0	–	4295	34.6	2.94	3.85	32.4
25	750	0.95	–	52	845	425	50.0	–	8314	23.3	2.78	0.00	22.2
25	750	0.96	–	49	845	425	50.0	–	4485	26.7	5.56	0.00	22.2
25	1000	0.92	–	82	294	210	14.6	–	12406	12.7	1.64	0.00	36.1
25	1000	0.95	–	83	286	205	18.4	–	11378	10.5	1.61	1.75	29.0
25	1000	0.96	–	104	49.0	20.6	17.7	–	10330	6.67	1.59	0.00	47.6
50	100	0.92	–	11	100	0.00	20	–	2805	80.0	0.00	0.00	0.00
50	100	0.95	–	1	6280	5133	40	–	2804	80.0	0.00	0.00	0.00
50	100	0.96	–	1	6280	5133	40	–	2803	80.0	0.00	0.00	0.00
50	250	0.92	–	1	80.0	0.00	40	–	2795	40.0	0.00	0.00	0.00
50	250	0.95	–	1	100	0.00	40	–	2796	60.0	0.00	0.00	0.00
50	250	0.96	–	1	100	0.00	40	–	2794	60.0	0.00	0.00	0.00
50	500	0.92	–	1	6280	2143	140	–	2773	100	0.00	0.00	21.4
50	500	0.95	–	1	6280	1993	160	–	2779	87.5	0.00	0.00	20.0
50	500	0.96	–	1	6280	1993	160	–	2763	87.5	0.00	0.00	20.0
50	750	0.92	–	8	3040	913	0.00	–	2947	230	6.45	0.00	3.20
50	750	0.95	–	10	3040	773	30.0	–	2942	177	0.00	0.00	22.2
50	750	0.96	–	8	3040	749	30.0	–	2955	185	0.00	0.00	18.9
50	1000	0.92	–	20	1470	398	40.0	–	2825	125	0.00	0.00	11.1
50	1000	0.95	–	13	1395	376	42.9	–	2880	127	3.03	0.00	28.8
50	1000	0.96	–	14	1327	355	59.1	–	2784	97.1	0.00	0.00	31.9

Table 2 Computational results, time limit 3000 seconds

9. The `ndosolver/fioracle` project, https://gitlab.com/frangio68/ndosolver_fioracle_project.
10. A. Virdis, G. Stea, and G. Nardini. Simulating lte/lte-advanced networks with simulte. In M.S. Obaidat, T. Ören, J. Kacprzyk, and J. Filipe, editors, *Simulation and Modeling Methodologies, Technologies and Applications*, volume 402 of *AISC*, pages 83–105. Springer, 2015.