Future Generation Computer Systems 110 (2020) 876–891

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Approaching the cell switch-off problem in 5G ultra-dense networks with dynamic multi-objective optimization

Francisco Luna [a,*], Pablo H. Zapata-Cano [a], Juan C. González-Macías [b], Juan F. Valenzuela-Valdés [c]

[a] *Dept. of Languages and Computer Science, Universidad de Málaga, Spain*
[b] *Dept. of Computer Systems and Telematics Engineering, Centro Universitario de Mérida, Universidad de Extremadura, Spain*
[c] *Dept. Signal Theory, Telematics and Communications, Universidad de Granada, Spain*

ARTICLE INFO

Article history:
Received 13 June 2019
Received in revised form 20 September 2019
Accepted 11 October 2019
Available online 17 October 2019

Keywords:
Dynamic multi-objective optimization
Problem-specific restart operator
Cell switch-off problem
Energy efficiency
5G networks

ABSTRACT

Energy efficiency is a major issue in the fifth generation (5G) of cellular networks as they require an ultra-dense deployment of small base stations (SBSs) to meet the forecasted traffic demands. Switching off cells is a widely recognized strategy to reduce the power consumption of these networks during off-peak conditions but, as times goes by, these demands change, thus requiring the activation or deactivation of a different set of cells that provide the users with a minimum QoS. In this context, the optimization problem of selecting which cells have to be switched on/off in each period of time has been approached from the dynamic multi-objective evolutionary (MOEA) domain, by proposing a novel restart method that enables the algorithms to react to changes in the traffic demands. The newly devised operator, named Adjacent Cell Restart (ACR), is based on exploiting the spatial continuity of the mobile users in the network. The experiments over a set of 9 ultra-dense networks with increasing densities of both users and base station has shown the enhanced capabilities of the ACR-enabled dynamic MOEAs to better approximate the newly induced the Pareto fronts in consecutive periods of time.

1. Introduction

The latest mobility reports published by Ericcson [1] and Cisco [2] are just confirming the predicted previsions about the increasing global mobile data traffic. The former, updated up to the first quarter (Q1) 2019, announces that such a traffic has grown by 82% between Q1 2018 and Q1 2019, reaching more than 28 exabytes per month and more than 6.0 billion mobile broadband subscriptions globally. On its part, Cisco has forecasted this amount of traffic to increase to 77 exabytes by 2022. Under this scenario, both public and private initiatives started to develop the fifth generation (5G) of cellular systems more than a decade ago. Through these years, three paradigms has been clearly identified to reach the challenging design requirements and expected performance indicators of 5G networks [3,4]: moving to millimeter wave (mmWave) spectrum to use larger bandwidths, enhancing spectral efficiency via multi-antenna transmission (massive, collaborative MIMO), and finally increasing spatial reuse through network densification [5]. This work focuses on the latest one.

In order to reduce the propagation loss of mmWave communications, improve the signal to interference plus noise ratio and reduce the latency, 5G networks require the deployment of a large number of small base stations (SBSs), which are close to the mobile users [5]. They are named ultra-dense networks (UDNs) [6,7]. However, these dense deployments come with a considerable increase in the power consumption of the system as SBSs are the most consuming device of the network (between 50% to 80%), regardless of its load [8]. This major issue could be addressed at different network levels [9], but one of the most promising one, to the point of being standarized by the 3rd Generation Partnership Project association [10], lies in switching off a number of SBSs when they are underutilized during low traffic demand periods. This is known as the Cell Switch-Off (CSO) problem [11]. CSO problems can be classified into two main categories [12]: online and offline CSO. Whereas online approaches are performed in real time (or in a few minutes basis), offline ones determines the set of active cells for longer time scales (from dozens of minutes to hours) in which the network conditions are assumed to be constant. Our work deals with offline CSO problems unless otherwise specified.

However, the decision problem that determines which subset of SBSs should be switched off in offline CSO is a NP-complete

[*] Corresponding author.
E-mail addresses: flv@lcc.uma.es (F. Luna), phzc@lcc.uma.es (P.H. Zapata-Cano), pablopadilla@ugr.es (J.C. González-Macías), juanvalenzuela@ugr.es (J.F. Valenzuela-Valdés).

https://doi.org/10.1016/j.future.2019.10.005
0167-739X/© 2019 Elsevier B.V. All rights reserved.

problem [13]. It also has to account not only for the energy consumption of the UDN, but also for any indicator of the QoS provided to the users. The CSO problem has also been addressed from different perspectives, ranging from clustering techniques [14,15] or game theory [16], to its formulation as an optimization problem [13]. As an optimization problem, CSO has been tackled with exact algorithms [17], simple heuristics [18,19], and both single [20] or multi-objective [21] metaheuristics. The present work extends a previous publication [22], in which the CSO problem has been formulated as a bi-objective optimization problem, where the power consumption of the UDN is to be minimized and the capacity provided to a set of users is to be maximized. This CSO problem has been addressed with two well known multi-objective evolutionary algorithms (MOEAs), NSGA-II and MOCell, which seek for a set of non-dominated solutions that approximates the Pareto front [23] of different instances.

In order to face the QoS optimization objective within the offline CSO framework, most of the existing works use a service demand that is considered also to be fairly constant during a given time span [13]. The optimization algorithm then searches for the optimal set of active SBSs that optimizes the problem objectives, and the next period is considered as an entirely new problem instance with a new service demand, which is addressed again from scratch. No information is passed on from one optimization stage to the next one. This is precisely one of the contributions of the present work: approaching the CSO problem from a dynamic multi-objective optimization perspective [24], in which the set of SBSs to switch-off for a given traffic pattern at period t is optimized, but starting from the search experience of the algorithm in the previous period $t-1$. The changes between consecutive periods have been modeled by users moving within the service area, which constitutes an actual challenge as network densification requires a frequent update of user association to the serving SBS [5,11]. We have named this new problem as the dynamic CSO (DCSO) problem.

When a multi-objective optimization problem changes with time, dynamic MOEAs have to detect when change occurs, and to react to this change to track the new Pareto front [25,26]. We consider here that the traffic demand will certainly change from one period to another, but remains constant within each of such periods, so the detection phase is trivial: it is just a true fact after changing the period. Once detection is done, dynamic MOEAs react to change by incorporating dynamic handling techniques, also known as restart methods [27]. Among them, diversity-based schemes try to maintaining a high diversity in the population so that the algorithms can easily approximate the new Pareto front. They were among the first proposed in the domain [28]. A second contribution of this work is the introduction of a novel diversity-based restart method specially tailored for the DCSO problem. It has been called Adjacent Cell Restart (ACR), and aims at exploiting the spatial continuity of the movements of the users in the network, by switching on deactivated SBSs, which are neighbors of cells that are already serving users, thus trying to anticipate the movement. In order to account for the user mobility, a third objective that measures the number of handovers induced by the new network configuration from one period to another has been included. The ACR operator has been incorporated to both NSGA-II and MOCell, and has been evaluated over a time horizon divided into 48 periods (e.g., a half-an-hour period over a full day), using 9 different UDN instances with different densities of up to 750 users and up to 3000 SBSs. To the best of our knowledge, the contributions of this work are:

1. This is the first time the (offline) CSO problem has been approached from the dynamic multi-objective evolutionary algorithm domain, not considering different periods of time as entirely new optimization phases.
2. A new restart operator specially engineered for the DCSO problem has been devised, showing very promising results over a wider variety of scenarios. Both NSGA-II and MOCell have been endowed with it, resulting in the so-called DNSGA-II-ACR and DMOCell-ACR algorithms. The behavior of the ACR operator has been fully characterized by evaluating 80 different configurations in a systematic experimental setup.
3. For comparison purposes, the same mechanism of DNSGA-II-A and DNSGA-II-B introduced in [28] have been implemented within the MOCell algorithm, resulting in the DMOCell-A and DMOCell-B. As a baseline algorithm, both NSGA-II and MOCell have been used to address each of the periods considered starting the search from scratch.

The work has been structured as follows. The next section discusses several related works that better contextualize the scientific contributions of this paper. Section 3 formally describes the model of the UDN used, as well as the formulation of the problem objectives for both the CSO and the DCSO problems. The ACR operator is fully described in Section 4, together with the dynamic MOEAs used in this work. Section 5 is devoted to detailing the experimentation carried out to assess the performance of the newly proposed restart operator. Finally, the main conclusions of the work as well as the lines for future research are included in Section 6.

2. Related works

As stated in the previous section, this work is focussed on offline CSO problems in 5G UDN networks. Attending to classification devised in [12], the actual scope of this work is an offline static coordination, with a non-uniform demand distribution and a site-based switch-off level. Given the NP-complete nature of the problem, and that the power consumption is usually in conflict with other QoS performance problem objectives [21], we are addressing the problem with multi-objective evolutionary algorithms. Unlike online CSO, it is considered that the UDN configuration is static during a long period of time, ranging from tens of minutes to hours and, to the best of our knowledge on the MOEAs related literature, when this period of time changes, a fairly new instance of the CSO problem is tackled restarting the search from scratch [13,22].

The most closely related work is presented by Chandhar and Sekhar Das [29], in which a framework for the dynamic optimization of OFDMA network (fourth generation, or 4G) is presented. The framework does not only address the activation/deactivation of base stations, but also configures other sectorized antenna parameters (power, tilting, etc.). It is a entire network planning problem. From a dynamic optimization point of view, the authors use a flavor of the NSGA-II algorithm that runs on a central controller and, based on gathering network information and daily traffic estimations, searches for an optimized network configuration. If estimations at a given period of the day match those of the previous day, the algorithm restarts from the same candidate solutions of the previous period. Otherwise, it randomly generates a number of individuals that replace the worst solutions in the population. As a 4G network, it is not highly densified, which is a major challenge in 5G.

Aissaoui Ferhi et al. [30] use a genetic algorithm (GA) for the optimization of a sectorized 5G network, but limited to only 12 base stations. Analogously to [29], the GA optimizes switching off sectors (not entire sites), as well as other antenna parameters. The problem objectives are aggregated on a single function, so no set of non-dominated solutions are identified. On the dynamism side, the work performs a robust optimization of the network,

Table 1
Summary of the advantages of this work with respect to the revised literature.

Ref.	5G	UDN	Multi-objective	Dynamic	Specialized operators
[29]	×	×	✓	✓	×
[13]	✓	✓	✓	×	×
[21]	✓	✓	✓	×	×
[30]	✓	×	×	✓	×
[22]	✓	✓	×	×	✓
This work	✓	✓	✓	✓	✓

that is, the objective function is evaluated over a traffic profile, thus searching for a solution that performs well on average.

A summary of the advantages of our approach with respect to the revised works is shown in Table 1, which analyzes the different works on the basis of the following five relevant features, one in each column: 5G, the CSO in 5G networks is addressed; UDN, attention is paid to highly densified networks; Multi-objective, a multi-objective formulation of the CSO problem is considered; Dynamic, it checks whether the CSO problem is addressed from a dynamic optimization perspective; and, finally, Specialized operators, which analyze if specialized search operators have been devised for the optimization problem and thus tailoring the optimizers. A "✓" symbols is used in the case a given work, cited in the first column, has considered the column feature, whereas "×" symbols appears otherwise. The table clearly highlights our contributions. Despite the rather large body of knowledge on the CSO problem, the novelty of our work is contrasted with the dynamic multi-objective approach of the CSO problem. We have considered a long time horizon composed of several of periods or epochs, as well as the use of mechanisms from the dynamic MOEA domain. The design of new problem-specific operators, capable of profiting from the search experience of past periods to better approximate the new Pareto fronts when the traffic demands change over time, has been also undertaken, a feature that is seldom reported in the literature.

3. The dynamic CSO problem

This section first introduces the modeling of the UDN and, then, it mathematically formulates the DCSO problem.

3.1. UDN modeling

This work considers a service area of 500 × 500 square meters, which has been discretized using a grid of 100 × 100 points (also called "pixels" or area elements), each covering a 25 m^2 area, where the signal power is assumed to be constant. Ten different regions have been defined with different propagation conditions. In order to compute the received power at each point, $P_{rx}[dBm]$, the following model has been used:

$$P_{rx}[dBm] = P_{tx}[dBm] + PLoss[dB] \tag{1}$$

where, P_{rx} is the received power in dBm, P_{tx} is the transmitted power in dBm, and $PLoss$ are the global signal losses, which depend on the given propagation region, and are computed as:

$$PLoss[dB] = GA + PA \tag{2}$$

where GA is the total gain of both antennas, and PA are the transmission losses in space, computed as:

$$PA[dB] = \left(\frac{\lambda}{2 \cdot \pi \cdot d}\right)^{K} \tag{3}$$

where d is the Euclidean distance to the SBS, K is the exponent loss, which ranges randomly in [2.0, 4.0] for each of the 10 different regions. The signal to interference plus noise ratio (SINR) for UE k, is computed as:

$$SINR_k = \frac{P_{rx,j,k}[mW]}{\sum_{i=1}^{M} P_{rx,i,k}[mW] - P_{rx,j,k}[mW] + P_n[mW]} \tag{4}$$

where $P_{rx,j,k}$ is the received power by UE k from SBS j, the summation is the total received power by UE k from all the SBSs operating at the same frequency that j, and P_n is the noise power, computed as:

$$P_n[dBm] = -174 + 10 \cdot \log_{10} BW_j \tag{5}$$

being BW_j the bandwidth of SBS j, defined as 5% of the SBS operating frequency (see Table 2). Finally, the capacity of the UE k is:

$$C_k^j[bps] = BW_k^j[Hz] \cdot \log_2(1 + SINR_k) \tag{6}$$

where BW_k^j is the bandwidth assigned to UE k when connected to SBS j, assuming a round robin scheduling, that is:

$$BW_k^j = \frac{BW_j}{N_j} \tag{7}$$

where N_j is the number of UEs connected to SBS j, and UEs are connected to the SBS with the highest SINR, regardless of its type.

In order to model an heterogeneous network, four different types of cells of decreasing size are considered: femtocells, picocells, microcells, and macrocells. Two subtypes of femto, pico, and microcells are also defined, summing up 7 cell types. Both the SBSs and the UEs are deployed using independent Poisson Point Processes (PPP) with different densities (defined by λ_P^{SBS} and λ_P^{UE}), respectively).

The power consumption of a transmitter is computed based on the model presented in [31], which considers that the device is transmitting over the fiber backhauling. Hence, the regular power consumption of SBS j, P_j, is expressed as:

$$P_j = \alpha \cdot P + \beta + \delta \cdot S + \rho \tag{8}$$

where P denotes the transmitted or radiated power of the transmitter, the coefficient α represents the efficiency of transmit power produced by an radio-frequency amplifier and feeder losses, the power dissipated owing to signal processing and site cooling is denoted by β, the dynamic power consumption per unit data is given by δ, being S the actual traffic demand served by the SBS, and, finally, the power consumption of the transmitting device is represented by the coefficient ρ.

The detailed parameterization of the scenarios addressed is included in Table 2, in which the column Eq. links the parameter to the corresponding equation in the formulation detailed above. The names in the last nine columns, XY, stand for the deployment densities of SBSs and UEs, respectively, so that X = {L, M, H}, meaning either low, medium, or high density deployments (λ_P^{SBS} parameter of the PPP), and Y = {L, M, H}, indicates a low, medium or high density of deployed UEs (λ_P^{UE} parameter of the PPP), in the last row of the table. The parameters G_{tx} and f of each type of cell refer to the transmission gain and the operating frequency (and its available bandwidth) of the antenna, respectively. Nine instances have been therefore used in this work in order to assess the performance of the newly designed restart operator.

3.2. The static CSO problem

Let $\mathcal{B}$ be the set of the SBSs randomly deployed. A solution to the CSO problem is a binary string $s \in \{0, 1\}^{|\mathcal{B}|}$, where s_i indicates whether SBS i is activated or not. The first objective to be minimized is therefore computed as:

$$\min f_{Power}(s) = \sum_{i=1}^{|\mathcal{B}|} s_i \cdot P_i \tag{9}$$

Table 2
Model parameters for cells and users.

Cell	Parameter	Eq.	LL	LM	LH	ML	MM	MH	HL	HM	HH
macro	G_{tx}	(2)	14								
	f	(5)	2 GHz (BW = 100 MHz)								
	α	(8)	21.45								
	β	(8)	344440								
	δ	(8)	2								
	$\rho[W]$	(8)	1								
micro1	G_{tx}	(2)	12								
	f	(5)	3.5 GHz (BW = 175 MHz)								
	α	(8)	15								
	β	(8)	10000								
	δ	(8)	1								
	$\rho[W]$	(8)	1								
	λ_P^{micro1} $[SBS/km^2]$		100	100	100	200	200	200	300	300	300
micro2	G_{tx}	(2)	10								
	f	(5)	5 GHz (BW = 250 MHz)								
	α	(8)	15								
	β	(8)	10000								
	δ	(8)	1								
	$\rho[W]$	(8)	1								
	λ_P^{micro2} $[SBS/km^2]$		100	100	100	200	200	200	300	300	300
pico1	G_{tx}	(2)	5								
	f	(5)	10 GHz (BW = 500 MHz)								
	α	(8)	9								
	β	(8)	6800								
	δ	(8)	0.5								
	$\rho[W]$	(8)	1								
	λ_P^{pico1} $[SBS/km^2]$		500	500	500	600	600	600	700	700	700
pico2	G_{tx}	(2)	7								
	f	(5)	14 GHz (BW = 700 MHz)								
	α	(8)	9								
	β	(8)	6800								
	δ	(8)	0.5								
	$\rho[W]$	(8)	1								
	λ_P^{pico2} $[SBS/km^2]$		500	500	500	600	600	600	700	700	700
femto1	G_{tx}	(2)	4								
	f	(5)	28 GHz (BW = 1400 MHz)								
	α	(8)	5.5								
	β	(8)	4800								
	δ	(8)	0.2								
	$\rho[W]$	(8)	1								
	λ_P^{femto1} $[SBS/km^2]$		1000	1000	1000	2000	2000	2000	3000	3000	3000
femto2	G_{tx}	(2)	3								
	f	(5)	66 GHz (BW = 3300 MHz)								
	α	(8)	5.5								
	β	(8)	4800								
	δ	(8)	0.2								
	$\rho[W]$	(8)	1								
	λ_P^{femto2} $[SBS/km^2]$		1000	1000	1000	2000	2000	2000	3000	3000	3000
UEs	λ_P^{UE} $[UE/km^2]$		1000	1000	1000	2000	2000	2000	3000	3000	3000

where P_i is the power consumption of SBS i (Eq. (8)).

Let $\mathcal{U}$ be the set of the UEs also deployed as described in the section above. In the static version of the CSO problem, the position of the UEs is constant and does not depend on the time. Subsequently, in order to compute total capacity of the system, UEs are first assigned to the active SBS that provides it with the highest SINR. Let $\mathcal{A}(s) \in \{0, 1\}^{|\mathcal{U}| \times |\mathcal{B}|}$ be the matrix where $a_{ij} = 1$ if $s_j = 1$ and SBS j serves UE i with the highest SINR, and $a_{ij} = 0$ otherwise. Then, the second objective to be maximized, which is the total capacity provided to all the UEs, is calculated as:

$$\max f_{Cap}(s) = \sum_{i=1}^{|\mathcal{U}|} \sum_{j=1}^{|\mathcal{B}|} s_j \cdot a_{ij} \cdot BW_i^j \tag{10}$$

where BW_i^j is the shared bandwidth of SBS j provided to UE i (Eq. (7)). We would like to remark that these two problem objectives are clearly conflicting one each other, as switching off base stations, that is, minimizing the power consumption of the network, will clearly decrease its capacity because the available bandwidth to serve users is reduced.

3.3. The DCSO problem

In this dynamic version of the CSO problem, the time comes into play, so the problem objectives defined for the static problems in the section above now depend on the period of time they are computed. This is one of the major contributions of our approach, as the existing related works that face the CSO problem consider that each of these periods is an entirely new instance, which is addressed independently by a MOEA [13]. We introduce here the concept of *epoch* within the CSO framework. An epoch is one of such a time interval of a predefined duration, which may range from dozens of minutes to few hours, in which the positions of UEs (i.e., the traffic demands) are assumed to be fixed. After an epoch is elapsed, the UEs move to a new position in the service area using the Random Waypoint Mobility model [32].

In our modeling, let $\mathcal{E} = \{e_1, e_2, \dots, e_t\}$ be a finite set of t consecutive epochs over a planning horizon (one day, one week, etc.). From the two objectives of the CSO problem, only $f_{Cap}(s)$ is affected by changes in the positions of the UEs, as $f_{Power}(s)$ only considers the set of SBSs switched-on (the tentative solution provided by the MOEAs). In order to formulate an epoch-dependent f_{Cap}, namely $f_{Cap}(s, e_t)$, a new definition of the matrix $\mathcal{A}$ is required: $\mathcal{A}^{\mathcal{E}}(s) \in \{0, 1\}^{|\mathcal{U}|\times|\mathcal{B}|\times|\mathcal{E}|}$ is the matrix where $a_{ijt} = 1$ if $s_j = 1$ SBS j serves UE i with the highest SINR at epoch t, and $a_{ijt} = 0$ otherwise. This capacity objective is now defined for a given epoch t as follows:

$$\max f_{Cap}(s,t) = \sum_{i=1}^{|\mathcal{U}|}\sum_{j=1}^{|\mathcal{B}|} s_j \cdot a_{ijt} \cdot BW_i^j \tag{11}$$

But the mobility of UEs also induces handovers. A handover is the transfer of a given UE from one SBS to another due to either its own movement or the switch-off of its serving SBS. This requires extra signaling from the access network, which is rather costly [13]. The DCSO problem formulation also considers this issue by measuring the induced handovers from one epoch to the next one:

$$\min f_{HO}(s,t) = \sum_{i=1}^{|\mathcal{U}|}\sum_{j=1}^{|\mathcal{B}|} s_j \cdot a_{ij(t-1)} \cdot (1 - a_{ijt}) \tag{12}$$

where $a_{ij(t-1)} \cdot (1 - a_{ijt}) = 1$ when SBS j served UE i at epoch $t - 1$, and, at epoch t, is served by a different SBS. If UE i is either not served by SBS j at $t - 1$, thus $a_{ij(t-1)} = 0$, or remains served by j at t, $a_{ijt} = 1$, this multiplication is 0, because no handover has been required.

In summary, the DCSO problem has to minimize the power consumption of the UDN network, maximize the capacity provided to the UEs, and minimizing the number of handovers induced by the new switching on plan for a new epoch.

4. The new restart operator

This section first details the ACR operator and, then, how it has been integrated within the dynamic MOEAs considered in this work. The source code developed for this work is available at https://bitbucket.org/pablozapata/udn/branch/FGFS.

4.1. The ACR operator

The rationale in the design of this operator is to exploit the locality and spatial continuity of the movements of users within the service area of the UDN. That is, if a given SBS j is active and does have any associated users, meaning that it is their serving SBS (Eq. (4)), the operator assumes that, if these users move somewhere, their new location will be close to that they are currently positioned. Therefore, the action of the ACR operator is to switch on the adjacent or neighboring SBSs of j with one main goal: enabling the algorithms to better track the approximated Pareto front by gradually activating SBSs in the areas where users may potentially move to. This introduces highly valuable genetic material into the solutions that will be passed on to the next epoch. Additionally, this strategy also provides the users with alternative SBSs to be associated with, thus balancing the load and increasing the capacity objective (Eq. (11)).

Algorithm 1 Pseudocode of the ACR operator

```
Input 1:  $P$         //A population of solutions
Input 2:  $r_p$       //The application rate on $P$
Input 3:  $r_{on}$    //The application rate on each SBS
Input 4:  $l_{on}$    //The length of the activation ratio
1: for $\forall p \in P$ do
2:   Select $p$ with probability $r_p$
3:   for $\forall b \in$ SBSwithUsersAssociated($p$) do
4:     for $\forall n \in$ Neigborhs($p$, $l_{on}$) do
5:       Select $n$ with probability $r_{on}$
6:       SwitchOn($n$)
7:     end for
8:   end for
9: end for
```

Algorithm 2 Generic dynamic multi-objective metaheuristic for the DCSO problem

```
1: $S(0) \leftarrow$ GenerateInitialSolutions()
2: $A(0) \leftarrow \emptyset$
3: for $e \in \mathcal{E}$ do
4:   Evaluation($S$, $e$)
5:   $A(0) \leftarrow$ Update($A(0)$, $S(0)$)
6:   $t \leftarrow 0$
7:   while not StoppingCriterion( ) do
8:     $t \leftarrow t + 1$
9:     $S(t) \leftarrow$ Variation($A(t-1)$, $S(t-1)$)
10:    Evaluate($S(t)$, $e$)
11:    $A(t) \leftarrow$ Update($A(t)$, $S(t)$)
12:  end while
13:  $S_e \leftarrow A(t)$
14:  $S \leftarrow ACR(A)$ //Reaction to change
15: end for
```

The pseudocode of the ACR operator is included in Algorithm 1. It is a randomized procedure that works on a population of solutions P, and does have three configuration parameters: r_p indicates the percentage of solutions from P that undergo to SBS activation, r_{on} determines the percentage of adjacent SBSs of a given SBS p that will be activated and, finally, l_{on} is the range of action of the operator for each SBS p, measured in terms of grid points of the discretized service area (see Section 3.1). The operator chooses a subset of solutions from P and then, for all the SBSs of each of these solutions having users associated (line 3), which could potentially move in the next epoch, it selects the adjacent SBSs within a range defined by l_{on} (function *Neighbors*(p, l_{on}) on line 4) and switches on those that are deactivated. The possibly modified set of solutions P is the output of the method.

4.2. Integration within dynamic MOEAs

Algorithm 2 displays a generic dynamic MOEA for solving the DCSO problem that has $\mathcal{E}$ epochs (line 3). As stated in the introduction, the definition of the DCSO problem addressed in this work considers that the traffic demand change from one epoch to another, and remains constant within each epoch, so no detection to change is required in the algorithms. The reaction to change is therefore undertaken at the end of the optimization process of the epoch (line 14), where the ACR operator is invoked. The algorithm records the best approximated front for each epoch (line 13).

The generic framework of Algorithm 2 has been particularized into two actual algorithms: NSGA-II [33] and MOCell [34]. They have been called DNSGA-II-ACR and DMOCell-ACR, respectively. These two algorithms have been chosen to build upon our previous work in [22]. This pseudocode can also embrace the DNSGA-II-A and DNSGA-II-B algorithms proposed in [28] (and their corresponding counterpart versions for MOCell: DMOCell-A and DMOCell-B), which have been used for comparison purposes.

5. Experimentation

This section is devoted to describing the experimentation conducted to assess the performance of the ACR operator. It first details the parameters used in all the algorithms, the methodology used to provide the results with statistical confidence is included next and, finally, the discussion and analysis of the results obtained. Both the algorithms and the DCSO problem have been implemented within the jMetal framework[1] and the source code is available for downloading at http://metanet5g.lcc.uma.es/files/fgcs.zip.

All the experiments have been executed in the facilities of the Supercomputing and Bioinformatics center of the Universidad de Málaga, named Picasso. It is an heterogeneous computing platform composed of a 48 nodes with 2 E5-2670 processors having 8 cores at 2.6 GHz and 64GB of RAM each, 7 shared memory machines with 2 TB of RAM each (7 nodes with 8 E7-4870 processors and 10 cores at 2.4 GHz), and 168 nodes with 2 Intel E5-2670 processors and 8 cores at 2.6 GHz and 32 GB of RAM each. The full hardware description can be found in http://www.scbi.uma.es/site/scbi/hardware.

5.1. Parameterization

All the algorithms in this work, NSGA-II and MOCell, either their static or dynamic versions, use a binary string representation for the solutions (see Section 3) and, as genetic operators, TwoPointCrossover with crossover rate of 1.0 and BitFlip mutation with a mutation rate of 1/L, being $L = |\mathcal{B}|$. The population size is always 100.

The ACR operator has been systematically evaluated over a total of 80 configurations, using the following set of values for its three parameters: $r_p = \{0.05, 0.25, 0.5, 0.75, 0.95\}$, $r_{on} = \{0.25, 0.5, 0.75, 1.0\}$, and $l_{on} = \{1, 2, 4, 8\}$ grid points, which is equivalent to say that the radius is $l_{on} = \{5, 10, 20, 40\}$, since the separation between the grid points is 5 m. These 80 configurations have been tested on both DNSGSA-II-ACR and DMOCell-ACR.

The four {DNSGA-II,DMOCell}-{A,B} versions also depend on a parameter $r_p = \{0.05, 0.25, 0.5, 0.75, 0.95\}$, which is the percentage of solutions involved in their restart method: whereas the A versions insert randomly generated solutions, the B ones replace existing solutions with randomly chosen mutated solutions.

Finally, a total number of 48 epochs have been considered in each run. This is aimed at simulating a time horizon of half-an-hour period in one day. Within each epoch, all the algorithms run for 25000 function evaluations, which is a rather common setting within the MOEA community. With the accurate problem modeling used, this setting of the dynamic MOEAs induces a runtime for a given epoch that depends on the size of the instances and, on average over all the independent runs performed, it ranges from 87.2 ± 48.8 and 118.6 ± 50.2 seconds in LL for DNSGA-II and DMOCell, respectively, to 285.8 ± 261.9 and 214.1 ± 189.1 seconds in HH. These figures clearly allow for an actual applicability of the approach, as the duration of an epoch (30 min) must account for two separated times: the optimization of the dynamic MOEAs for the current traffic data, and the actual deployment of the solution in the UDN, i.e., switching on/off the base stations, which requires roughly 30 seconds [35]. Whereas the latter is a technological constraint imposed by the SBS manufacturer, the former is on our side, and we believe that the reported times clearly fit in the assumed timeframe. Besides, it is also important to remark that these runtimes have been reached within the multi-core computing nodes of the Picasso supercomputer, which are shared among different independent runs of the dynamic MOEAs and even among processes from different users. If dedicated resources would have been used, these runtimes would be clearly reduced. And, lastly, MOEAs are clearly prone to parallelism [36], an advanced tool that may aid to shorten the executions times drastically, if a tighter timeframe is imposed.

5.2. Methodology

In order to guarantee a fair comparison among the algorithms, the random UDN generation (deployment of base stations) is undertaken on the basis of the same 30 seeds, thus ensuring that all the algorithms face the same set of instances in the 30 independent runs required to gather statistical data. These 30 independent runs have been performed for a total of 91 different configurations: 80 of the ACR operator, 5 + 5 for the A and B versions, and 1 more for the algorithms without any sort of reaction to change, that is, the isolated optimization of each epoch. This is done for the 9 instances defined in Table 2, which means $91 \times 30 \times 9 \times 2 = 49140$ runs. The mobility of the users in the network is the same for all the cases. In summary, the total accumulated CPU time for all these experiments required more than 12.7 years. We have collected the approximated fronts and the approximated Pareto optimal sets with the problem variables for all the epochs of all the algorithms, resulting also in more than 300 GB of data.

In order to measure the quality of all these approximated fronts, we have used the Hypervolume indicator [37], which is recognized as one of the most suitable Pareto-compliant metric in the multi-objective community. Higher values of HV are better. However, this indicator is not free from an arbitrary scaling of the objectives, so we have composed a reference Pareto front for normalizing each approximated front prior to compute the HV value. The point here is that the reference Pareto front does not only depends on the UDN configuration, but also on the epoch. Indeed, from a strict point of view, each epoch induces a different problem instance. As a consequence, we have 9 × 48 reference Pareto fronts, which are built by retrieving the non-dominated solutions for all the algorithms on a given UDN configuration and a given epoch.

5.3. Results

This section has been further structured into different parts to better organize the analysis of the results. Given the amount of data, we first give a big picture on the impact of the ACR operator in the search of the algorithms over all the epochs. Then, we develop a by-epoch and by-epoch plus by-algorithm analysis, and, finally, we compare our proposal to existing dynamic MOEAs from the literature.

[1] https://github.com/jMetal/jMetal

Table 3
HV values aggregated over all the epochs and by configurations of the ACR operator in the LL, LM and LH instances.

	LL		LM		LH	
	DNSGA-II	DMOCell	DNSGA-II	DMOCell	DNSGA-II	DMOCell
NRC	0.1078$_{\pm0.0114}$	0.1173$_{\pm0.0124}$	0.1094$_{\pm0.0116}$	0.1180$_{\pm0.0125}$	0.1117$_{\pm0.0118}$	0.1191$_{\pm0.0126}$
$r_p = 0.05$	0.2398$_{\pm0.0935}$	0.2650$_{\pm0.1032}$	0.2437$_{\pm0.0949}$	0.2661$_{\pm0.1036}$	0.2487$_{\pm0.0968}$	0.2697$_{\pm0.1048}$
$r_p = 0.25$	0.2645$_{\pm0.1029}$	0.2925$_{\pm0.1137}$	0.2668$_{\pm0.1039}$	0.2951$_{\pm0.1147}$	0.2719$_{\pm0.1057}$	0.2995$_{\pm0.1163}$
$r_p = 0.50$	0.2739$_{\pm0.1065}$	0.3121$_{\pm0.1212}$	0.2760$_{\pm0.1073}$	0.3146$_{\pm0.1222}$	0.2815$_{\pm0.1094}$	0.3201$_{\pm0.1241}$
$r_p = 0.75$	0.2797$_{\pm0.1088}$	0.3280$_{\pm0.1274}$	0.2820$_{\pm0.1096}$	0.3313$_{\pm0.1287}$	0.2874$_{\pm0.1116}$	0.3381$_{\pm0.1311}$
$r_p = 0.95$	0.2836$_{\pm0.1102}$	0.3352$_{\pm0.1303}$	0.2860$_{\pm0.1112}$	0.3390$_{\pm0.1318}$	0.2914$_{\pm0.1132}$	0.3470$_{\pm0.1347}$
$r_{on} = 0.25$	0.2691$_{\pm0.1143}$	0.3080$_{\pm0.1310}$	0.2713$_{\pm0.1151}$	0.3111$_{\pm0.1323}$	0.2768$_{\pm0.1174}$	0.3165$_{\pm0.1344}$
$r_{on} = 0.50$	0.2683$_{\pm0.1139}$	0.3065$_{\pm0.1303}$	0.2709$_{\pm0.1150}$	0.3094$_{\pm0.1315}$	0.2759$_{\pm0.1170}$	0.3151$_{\pm0.1338}$
$r_{on} = 0.75$	0.2679$_{\pm0.1138}$	0.3061$_{\pm0.1301}$	0.2703$_{\pm0.1147}$	0.3084$_{\pm0.1311}$	0.2754$_{\pm0.1168}$	0.3139$_{\pm0.1333}$
$r_{on} = 1.00$	0.2679$_{\pm0.1137}$	0.3056$_{\pm0.1299}$	0.2699$_{\pm0.1146}$	0.3079$_{\pm0.1310}$	0.2754$_{\pm0.1168}$	0.3141$_{\pm0.1333}$
$l_{on} = 1$	0.2713$_{\pm0.1152}$	0.3121$_{\pm0.1328}$	0.2736$_{\pm0.1162}$	0.3149$_{\pm0.1340}$	0.2791$_{\pm0.1183}$	0.3207$_{\pm0.1364}$
$l_{on} = 2$	0.2709$_{\pm0.1151}$	0.3117$_{\pm0.1326}$	0.2732$_{\pm0.1160}$	0.3136$_{\pm0.1335}$	0.2788$_{\pm0.1182}$	0.3200$_{\pm0.1360}$
$l_{on} = 4$	0.2694$_{\pm0.1144}$	0.3070$_{\pm0.1304}$	0.2709$_{\pm0.1150}$	0.3100$_{\pm0.1317}$	0.2763$_{\pm0.1172}$	0.3152$_{\pm0.1338}$
$l_{on} = 8$	0.2617$_{\pm0.1110}$	0.2955$_{\pm0.1253}$	0.2646$_{\pm0.1122}$	0.2984$_{\pm0.1265}$	0.2691$_{\pm0.1141}$	0.3037$_{\pm0.1286}$

Table 4
HV values aggregated over all the epochs and by configurations of the ACR operator in the ML, MM and MH instances.

	ML		MM		MH	
	DNSGA-II	DMOCell	DNSGA-II	DMOCell	DNSGA-II	DMOCell
NRC	0.0765$_{\pm0.0082}$	0.0981$_{\pm0.0104}$	0.0745$_{\pm0.0080}$	0.0961$_{\pm0.0102}$	0.0755$_{\pm0.0081}$	0.0971$_{\pm0.0103}$
$r_p = 0.05$	0.2702$_{\pm0.1051}$	0.3040$_{\pm0.1181}$	0.2701$_{\pm0.1051}$	0.3061$_{\pm0.1188}$	0.2731$_{\pm0.1064}$	0.3082$_{\pm0.1197}$
$r_p = 0.25$	0.2931$_{\pm0.1140}$	0.3348$_{\pm0.1298}$	0.2929$_{\pm0.1138}$	0.3360$_{\pm0.1301}$	0.2965$_{\pm0.1153}$	0.3387$_{\pm0.1313}$
$r_p = 0.50$	0.3016$_{\pm0.1172}$	0.3592$_{\pm0.1392}$	0.3021$_{\pm0.1173}$	0.3594$_{\pm0.1391}$	0.3053$_{\pm0.1187}$	0.3625$_{\pm0.1404}$
$r_p = 0.75$	0.3077$_{\pm0.1196}$	0.3808$_{\pm0.1476}$	0.3078$_{\pm0.1195}$	0.3808$_{\pm0.1474}$	0.3108$_{\pm0.1208}$	0.3842$_{\pm0.1488}$
$r_p = 0.95$	0.3118$_{\pm0.1211}$	0.3875$_{\pm0.1505}$	0.3118$_{\pm0.1210}$	0.3865$_{\pm0.1499}$	0.3155$_{\pm0.1226}$	0.3902$_{\pm0.1514}$
$r_{on} = 0.25$	0.2973$_{\pm0.1260}$	0.3544$_{\pm0.1504}$	0.2974$_{\pm0.1260}$	0.3557$_{\pm0.1508}$	0.3006$_{\pm0.1275}$	0.3589$_{\pm0.1522}$
$r_{on} = 0.50$	0.2966$_{\pm0.1257}$	0.3537$_{\pm0.1501}$	0.2967$_{\pm0.1257}$	0.3539$_{\pm0.1500}$	0.2995$_{\pm0.1271}$	0.3571$_{\pm0.1514}$
$r_{on} = 0.75$	0.2965$_{\pm0.1258}$	0.3528$_{\pm0.1497}$	0.2963$_{\pm0.1255}$	0.3535$_{\pm0.1498}$	0.3004$_{\pm0.1275}$	0.3560$_{\pm0.1510}$
$r_{on} = 1.00$	0.2960$_{\pm0.1255}$	0.3521$_{\pm0.1494}$	0.2961$_{\pm0.1255}$	0.3520$_{\pm0.1491}$	0.2991$_{\pm0.1268}$	0.3550$_{\pm0.1505}$
$l_{on} = 1$	0.3004$_{\pm0.1274}$	0.3602$_{\pm0.1531}$	0.3002$_{\pm0.1272}$	0.3614$_{\pm0.1533}$	0.3035$_{\pm0.1288}$	0.3642$_{\pm0.1546}$
$l_{on} = 2$	0.2997$_{\pm0.1271}$	0.3596$_{\pm0.1527}$	0.2998$_{\pm0.1270}$	0.3598$_{\pm0.1526}$	0.3036$_{\pm0.1288}$	0.3630$_{\pm0.1541}$
$l_{on} = 4$	0.2975$_{\pm0.1261}$	0.3541$_{\pm0.1501}$	0.2977$_{\pm0.1261}$	0.3540$_{\pm0.1499}$	0.3008$_{\pm0.1276}$	0.3574$_{\pm0.1515}$
$l_{on} = 8$	0.2889$_{\pm0.1223}$	0.3391$_{\pm0.1435}$	0.2888$_{\pm0.1222}$	0.3399$_{\pm0.1437}$	0.2917$_{\pm0.1235}$	0.3425$_{\pm0.1449}$

Table 5
HV values aggregated over all the epochs and by configurations of the ACR operator in the HL, HM and HH instances.

	HL		HM		HH	
	DNSGA-II	DMOCell	DNSGA-II	DMOCell	DNSGA-II	DMOCell
NRC	0.0181$_{\pm0.0024}$	0.0472$_{\pm0.0051}$	0.0158$_{\pm0.0020}$	0.0456$_{\pm0.0050}$	0.0138$_{\pm0.0020}$	0.0424$_{\pm0.0047}$
$r_p = 0.05$	0.2528$_{\pm0.0987}$	0.2944$_{\pm0.1142}$	0.2571$_{\pm0.1003}$	0.3001$_{\pm0.1165}$	0.2525$_{\pm0.0984}$	0.2959$_{\pm0.1147}$
$r_p = 0.25$	0.2719$_{\pm0.1060}$	0.3183$_{\pm0.1234}$	0.2775$_{\pm0.1082}$	0.3239$_{\pm0.1256}$	0.2727$_{\pm0.1062}$	0.3192$_{\pm0.1237}$
$r_p = 0.50$	0.2799$_{\pm0.1090}$	0.3382$_{\pm0.1312}$	0.2848$_{\pm0.1109}$	0.3445$_{\pm0.1337}$	0.2802$_{\pm0.1090}$	0.3391$_{\pm0.1314}$
$r_p = 0.75$	0.2857$_{\pm0.1112}$	0.3582$_{\pm0.1390}$	0.2910$_{\pm0.1133}$	0.3641$_{\pm0.1414}$	0.2858$_{\pm0.1111}$	0.3579$_{\pm0.1389}$
$r_p = 0.95$	0.2891$_{\pm0.1126}$	0.3615$_{\pm0.1406}$	0.2949$_{\pm0.1148}$	0.3668$_{\pm0.1427}$	0.2894$_{\pm0.1125}$	0.3601$_{\pm0.1400}$
$r_{on} = 0.25$	0.2760$_{\pm0.1173}$	0.3358$_{\pm0.1424}$	0.2815$_{\pm0.1196}$	0.3413$_{\pm0.1449}$	0.2761$_{\pm0.1172}$	0.3359$_{\pm0.1424}$
$r_{on} = 0.50$	0.2755$_{\pm0.1171}$	0.3342$_{\pm0.1417}$	0.2809$_{\pm0.1194}$	0.3404$_{\pm0.1445}$	0.2762$_{\pm0.1172}$	0.3346$_{\pm0.1419}$
$r_{on} = 0.75$	0.2757$_{\pm0.1172}$	0.3334$_{\pm0.1414}$	0.2803$_{\pm0.1191}$	0.3392$_{\pm0.1439}$	0.2756$_{\pm0.1170}$	0.3341$_{\pm0.1416}$
$r_{on} = 1.00$	0.2752$_{\pm0.1170}$	0.3332$_{\pm0.1413}$	0.2804$_{\pm0.1191}$	0.3386$_{\pm0.1437}$	0.2752$_{\pm0.1168}$	0.3332$_{\pm0.1412}$
$l_{on} = 1$	0.2793$_{\pm0.1187}$	0.3414$_{\pm0.1449}$	0.2843$_{\pm0.1208}$	0.3473$_{\pm0.1475}$	0.2789$_{\pm0.1184}$	0.3415$_{\pm0.1449}$
$l_{on} = 2$	0.2784$_{\pm0.1184}$	0.3401$_{\pm0.1443}$	0.2836$_{\pm0.1205}$	0.3454$_{\pm0.1466}$	0.2785$_{\pm0.1182}$	0.3401$_{\pm0.1442}$
$l_{on} = 4$	0.2766$_{\pm0.1175}$	0.3341$_{\pm0.1416}$	0.2818$_{\pm0.1197}$	0.3399$_{\pm0.1441}$	0.2769$_{\pm0.1176}$	0.3348$_{\pm0.1418}$
$l_{on} = 8$	0.2681$_{\pm0.1138}$	0.3210$_{\pm0.1358}$	0.2734$_{\pm0.1161}$	0.3269$_{\pm0.1385}$	0.2689$_{\pm0.1140}$	0.3215$_{\pm0.1360}$

5.3.1. ACR characterization

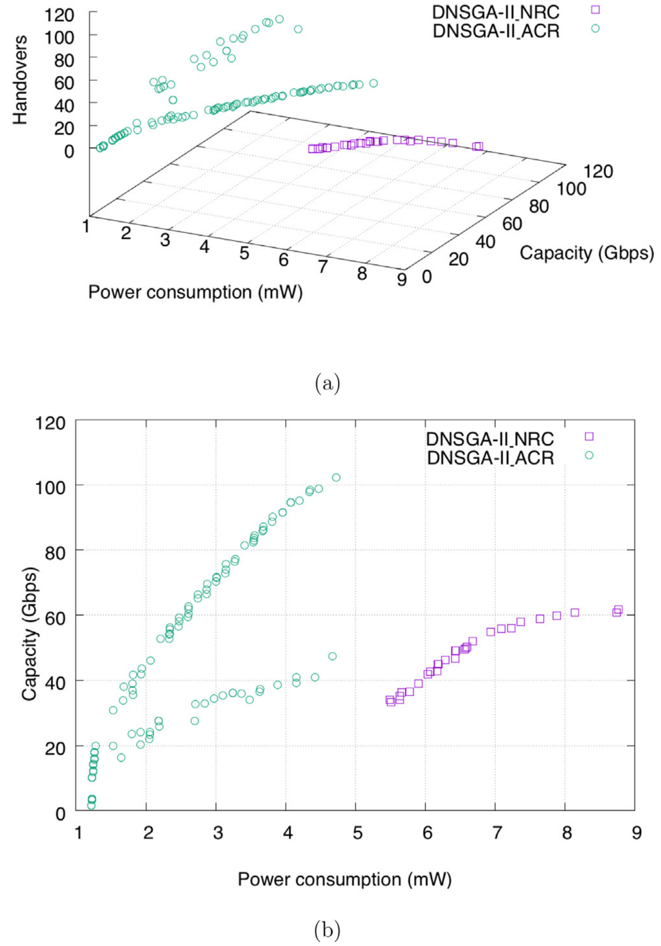
In order to evaluate the effect of the different configuration parameters of the ACR operator, we have aggregated the HV value of a given setting, over all the defined ones in the previous section, and over all the 48 epochs. That is, the row $r_p = 0.05$ shows the average value of the HV over all the configurations out of the 80 defined above in which $r_p = 0.05$. The goal is to characterize the impact of each setting separately. Tables 3–5 include these aggregated HV values for DNSGA-II-ACR and DMOCell-ACR corresponding to the L*, M*, and H* instances, respectively. The first row in the tables, named NRC or Not Reaction to Change, shows the results of the standard NSGA-II and MOCell that optimizes each epoch from scratch.

Let us try to shed some light over such an amount of data. The first conclusion to be highlighted is that any ACR-enable dynamic algorithm outperform the NRC versions. This means that the genetic material that the ACR operator passes on to the next epoch is very helpful to approximate the newly induced Pareto front defined by the new users positions. Given the normalization procedure detailed in the previous section, differences in the HV values are significant and increase with highly dense scenarios. Indeed, for the L*, M*, and H* instances, the HV value of NRC versions is roughly 30%, 25% and 10% of that of the ACR versions, respectively. In order to better illustrate the impact of the newly devised restart operator, Figs. 1 and 2 display two representative cases of approximated Pareto fronts reached by DNSGA-II-ACR and DMOCell-ACR for a given epoch of the HH instance, respectively. Fig. 1a depicts a 3D plot with the trade-off solutions for the DCSO problem (three objectives) of DNSGA-II-ACR and DNSGA-II-NRC, clearly showing that the former has computed a better approximated front with non-dominated solutions having lower power consumption and higher capacity. A projection of this 3D front on the consumption *vs.* capacity plane (Fig. 1b) better shown this fact, where even DNSGA-II-ACR solutions with a greater number of handovers always dominate those of DNSGA-II-NRC. Reaching approximated Pareto fronts in which all non-dominated solutions has zero handovers is the second representative case, and is very frequent in DMOCell-ACR. Fig. 2 includes a typical approximated front (2D) reached by this dynamic MOEA. Again, it can be seen that the ACR operator has clearly improved the search of the algorithm by incorporating information from the previous epoch, rather than starting the exploration from scratch as the NRC versions do. At this point, it is important to remark that this scenario, where no handovers are necessary, is a feasible scenario as users may move in the network and still being served by the same base station, if it remains on the coverage radio that receives the signal with the higher SINR (Eq. (4)). Therefore, the dynamic MOEAs then seek to switch-off other base stations of the network, reducing both the power consumption and the interferences, thus increasing the SINR and therefore the capacity provided to the users. The ACR operator is able to pass this information between consecutive epochs. Finally, as to the comparison between the dynamic algorithms DNSGA-II and DMOCell, the results are consistent with those reached in [22], the latter always outperforms the former, also with larger differences as the densification of the instances increases.

If we analyze the results by the ACR parameter, it can be seen that the most decisive one is r_p. In the corresponding block of this parameter in all the three tables, the HV values of both DNSGA-II and DMOCell steadily increase with higher values of r_p. That is, when larger portions of the final set of non-dominated solutions at epoch e are passed on to epoch $e + 1$, the two algorithms are able to better react to the change in the traffic demand, and thus reaching better approximations to the Pareto fronts. This enhancement in the search capabilities of the algorithms is specially relevant for DCMOCell, which typically increases the HV value almost in 0.8 from $r_p = 0.05$ to $r_p = 0.95$ in all the nine instances, whereas DNSGA-II does only in 0.3.

The value of parameter r_{on} is much less important on the behavior of the ARC versions of the algorithm. The second block of results in Tables 3–5 points out that differences in the HV values are very tight in all the aggregated executions. This is consistent in both DNSGA-II and DMOCell. Recall that this parameter regulates the percentage of deactivated neigbors of a given base station that are switched on. This could be explained by the high density of deployed SBSs in all the considered instances, and the overlapping among the different neighborhoods, which ends up propagating a generalized activation of SBSs in localized regions of the networks where a traffic demand (users) exists.

Finally, the impact of l_{on} follows a different trend, since the aggregated HV values get decreased when the value of this parameter gets bigger. Indeed, except for only one exception (DNSGA-II in the MH instance on Table 4), it holds that $HV^{l_{on}=1} > HV^{l_{on}=2} > HV^{l_{on}=4} > HV^{l_{on}=8}$. Even though differences between the three first values of l_{on} are negligible, the average HV value reported by all the configurations with $l_{on} = 8$ is clearly worse (lower). The interpretation in terms of the functioning of the ACR operator is that it does not make much sense to enlarge the range of activation of neighboring cells, because of the overlapping described in the previous paragraph. With higher values of this parameter, the operator activates very large numbers of SBSs in the UDN, driving the algorithms in the next epoch to restart the search from an



(a)

(b)

Fig. 1. Representative approximated fronts of a given epoch for the HH instance reached by DNSGA-II with both the ACR and NRC versions. The subfigure (a) corresponds to the actual 3D front, whereas (b) is the projection on the power consumption *vs.* capacity plane.

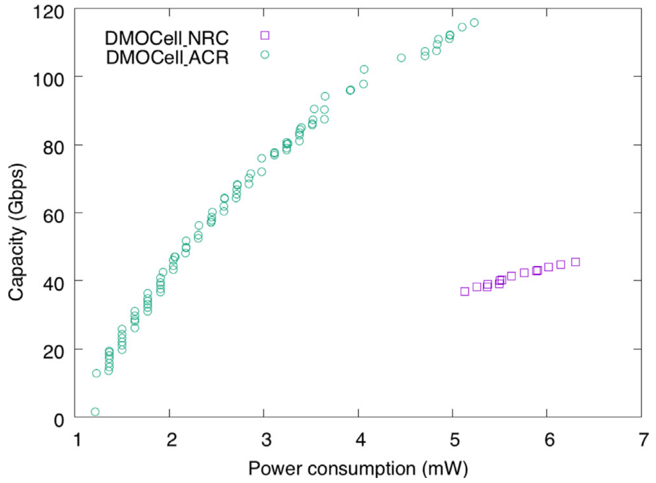
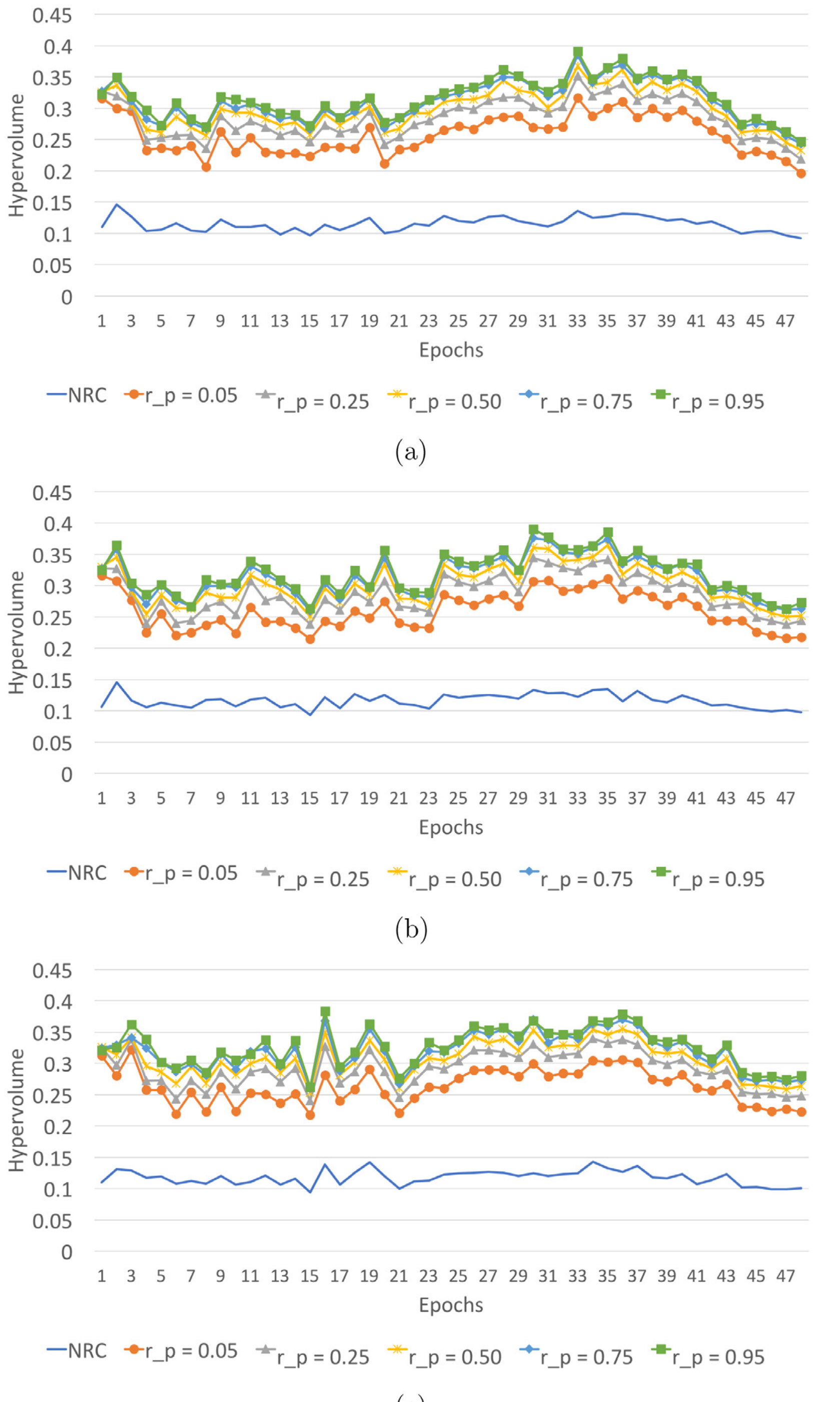


Fig. 2. Representative approximated fronts of a given epoch for the HH instance reached by DMOCell with both the ACR and NRC versions. It is a 2D projection on the power consumption *vs.* capacity plane as the number of handovers in all the non-dominated solutions of the two algorithms is zero.



(a)

(b)

(c)

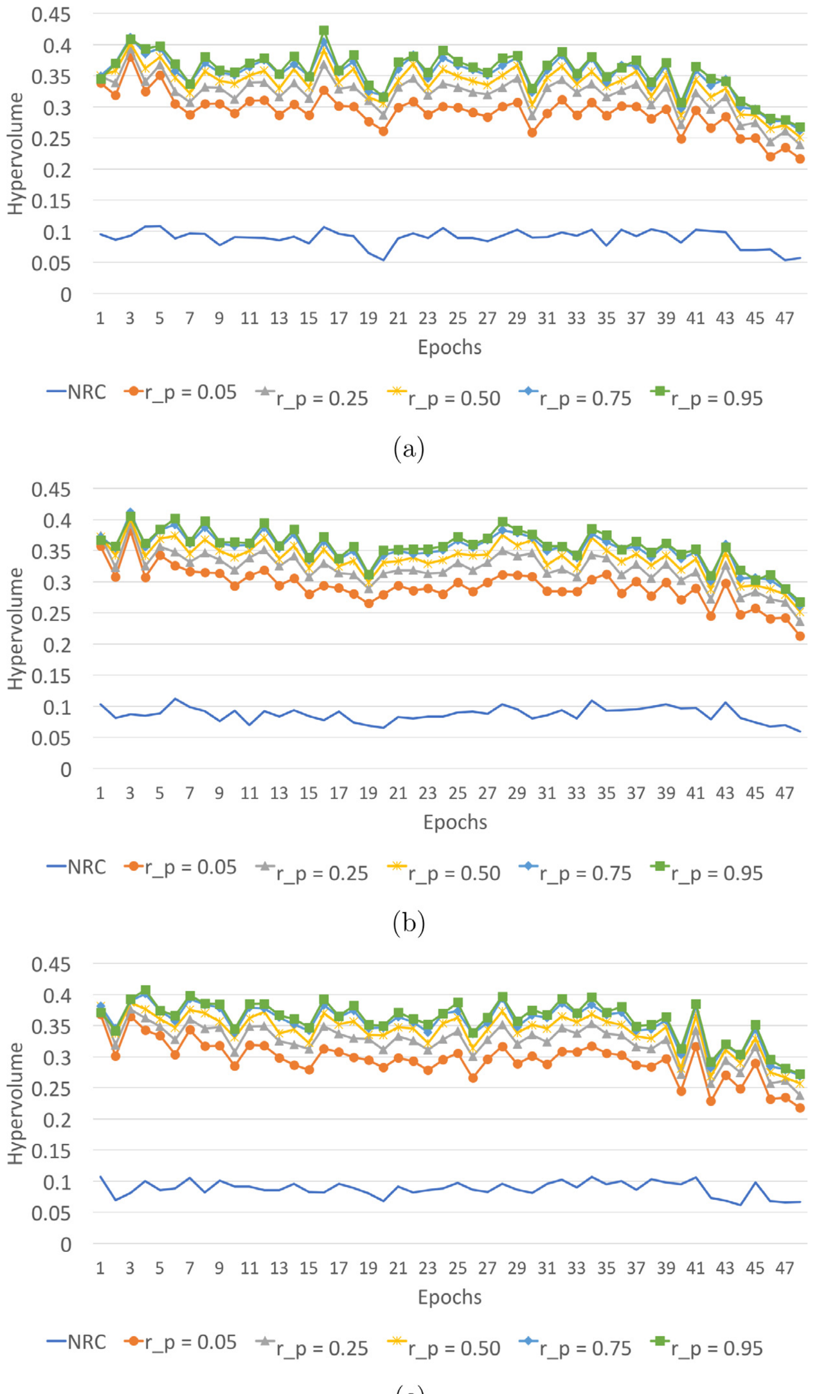
Fig. 3. Average HV values by epoch over all the ACR configurations of the (a) LL, (b) LM, and (c) LH scenarios. The different lines aggregate the values of the configurations with $r_p = \{0.05, 0.25, 0.50, 0.75, 0.95\}$. The NRC results are included as the baseline. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

network operating at full power. In the end, larger values of l_{on} do not allow to properly track the mobility of the users for the next epoch.

5.3.2. Analysis by epoch

In this section, we further analyze the results of the ACR-enable dynamic MOEAs, but now considering their behavior throughout the predefined 48 epochs. Starting upon the conclusions of the previous section, that have shown that the r_p parameter of the ACR operator is the one with a major impact in the quality of the approximated Pareto fronts, we again have aggregated the HV values over all the configurations that maintain r_p fixed. Figs. 3, 4, and 5 include the evolution of these HV values over the 48 epochs, together with that of the NRC version.

There are several findings one can identify across all the figures: first, the advantage of the configuration with $r_p = 0.95$ to track the Pareto front of each epoch is kept during all the time horizon. As it can be seen, the green lines with squares always plot on the upper part with the higher (better) HV values. This clearly supports the values in Tables 3–5 of the previous



(a)

(b)

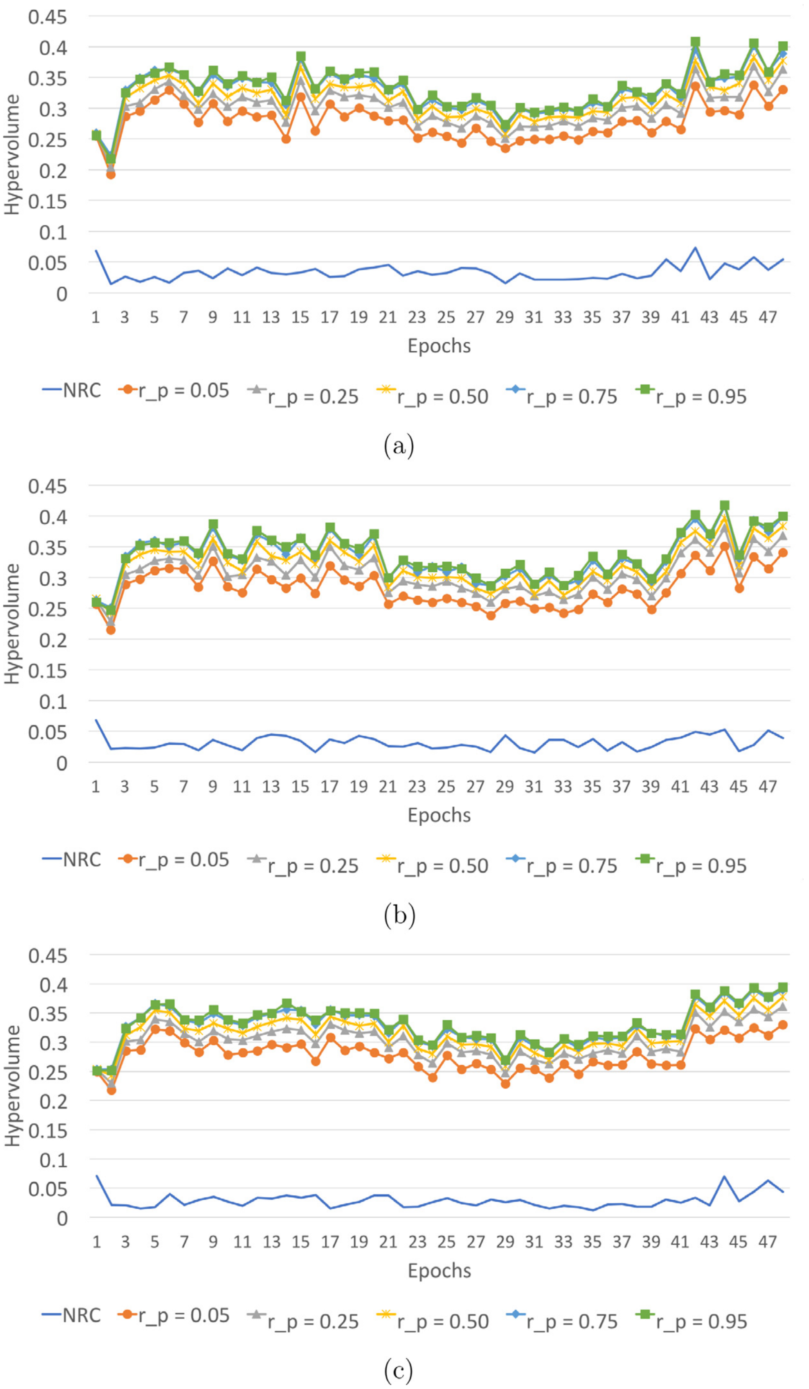
(c)

Fig. 4. Average HV values by epoch over all the ACR configurations of the (a) ML, (b) MM, and (c) MH scenarios. The different lines aggregate the values of the configurations with $r_p = \{0.05, 0.25, 0.50, 0.75, 0.95\}$. The NRC results are included as the baseline. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

section. Second, there also exist epochs in which differences are negligible (epoch 3 in all the instances), meaning that the new traffic demand for the epoch induces a new Pareto front which is easily tracked regardless of the configuration. On the other hand, other epochs exist where the gap between the configuration with $r_p = 0.95$ and the four remaining ones is significant. In any case, there is not a clear pattern across all the instances as the changes in the traffic demand may affect differently depending on the actual deployment of the base stations. Third, it is also important to remark that the mobility of the users, i.e., the change in traffic demands throughout the epochs, provokes strong updates in the newly induced Pareto fronts, which is reflected by the peaks one can find in consecutive periods. And, finally, these figures also show the actual, important need of using dynamic MOEAs to address the CSO problem rather than optimizing independently each epoch. Indeed, the NRC line that corresponds to these settings clearly reports the worst HV value.

5.3.3. Analysis by-epoch and by-algorithm

This is the first section in which we discuss the impact of the ACR operator separately on DNSGA-II and DMOCell. Again, given the number of different configurations, we have processed



(a)

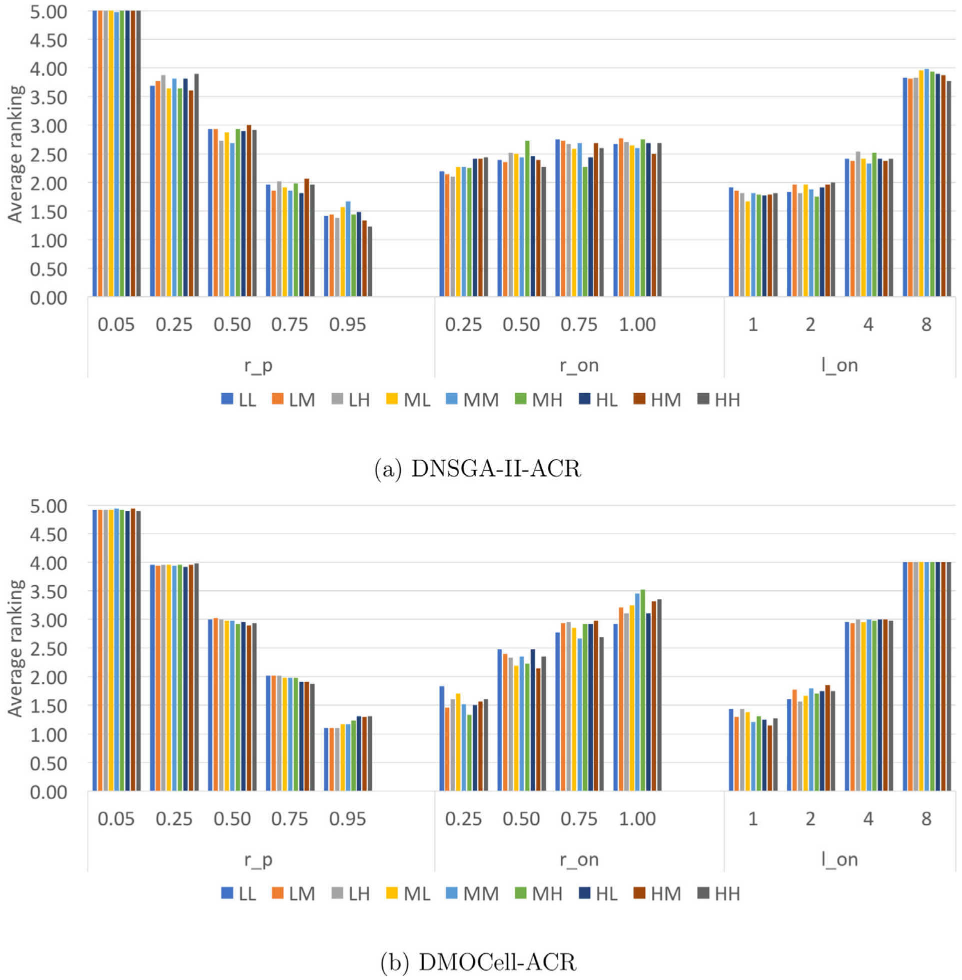
(b)

(c)

Fig. 5. Average HV values by epoch over all the ACR configurations of the (a) HL, (b) HM, and (c) HH scenarios. The different lines aggregate the values of the configurations with $r_p = \{0.05, 0.25, 0.50, 0.75, 0.95\}$. The NRC results are included as the baseline. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the results in order to compact them properly. In this case, we have ranked the configurations of r_p, r_{on} and l_{on} separately for the two algorithms and computed the average ranking over all the epochs. The resulting values for each of the nine UDN scenarios are displayed in Fig. 6. Note that, in this figures, the lower the average ranking, the better, as the configuration that scores the best (the highest HV) is assigned a rank of 1.

The first issue that clearly raises from the figures is that the most determining parameter in the two algorithms is r_p, a consistent fact with the previous findings based on aggregated values. The larger the value of r_p, the better (lower) the rank in all the nine UDN scenarios. However, the impact of r_{on}, which has shown to be slightly decisive in Section 5.3.1, has become relevant within the DCMOCell-ACR algorithm. Indeed, it can be seen in Fig. 6.b that smaller values of r_{on} have better rankings. Within the DNSGA-II-ACR framework, this tendency is not that well defined and one can find actual counterexamples, such as the MH instance that has a better rank for $r_{on} = 0.75$ that for $r_{on} = 0.5$. Finally, the results for l_{on} are on the same line as discussed above, extending the effect of the ACR operator to deactivated base stations that are further than nearby locations of the network clearly scores the worst. Small differences are found in DNSGA-II-ACR and DMOCell-ACR, as the ranks shows that $l_{on} = 1$ is the best setting for this parameter in the latter,



(a) DNSGA-II-ACR

(b) DMOCell-ACR

Fig. 6. Average HV-based ranking of the different configurations of the ACR-enabled versions of (a) DNSGA-II and (b) DMOCell. They are grouped into three sections, accounting for the ACR governing parameters r_p, r_{on}, and $l_o n$, respectively.

Table 6
Best ACR settings for each UDN scenario.

ACR	r_p	r_{on}	l_{on}
LL	0.95	0.75	1
LM	0.95	0.50	1
LH	0.95	0.25	1
ML	0.95	0.50	1
MM	0.95	0.50	1
MH	0.95	0.50	1
HL	0.95	1.00	1
HM	0.95	0.50	1
HH	0.95	0.50	1

whereas in DNSGA-II-ACR both $l_{on} = 1$ and $l_{on} = 2$ provide similar effectiveness.

5.4. Comparison with other dynamic MOEAs

The final analysis of the results now targets actual algorithms, not averaging over a set of different configurations. The comparison basis is defined now by, first, the NRC versions of DNSGA-II and DMOCell, which do no react to changes, and, second, the A and B versions proposed in [28]. Two different configurations of DNSGA-II-ACR and DMOCell-ACR has been included in the comparison: the configuration that perform the best considering the 9 UDN scenarios, which uses $r_p = 0.95$, $r_{on} = 0.5$, and $l_{on} = 1$, and a final comparison with the best algorithm for each instance. Its parameters are included in Table 6. The average HV values across all the 48 epochs of all these algorithms are included in Tables 7, 8, and 9, where a gray colored background has been used to highlight the algorithm that reported the best (higher) HV value.

The main conclusion that can be drawn from these tables is that the ACR versions of the algorithms have always better approximated the Pareto fronts of the 9 considered instances (higher HV values). Differences are significant with respect to the A version, and a bit tighter in the B versions, in which the configuration with $r_p = 0.95$ has also performed quite well. In order to better show the benefits of the ACR operator, Fig. 7 displays the gap between the HV value of the NRC version, which acts as a comparing baseline, and the different dynamic MOEAs that incorporates a reaction-to-change mechanism. It can clearly seen that the two last columns in each group, which correspond to the ACR-enable versions of the algorithms, have the larger differences. Given the normalization procedure followed to compute the HV values, these minimal differences induces substantial changes in the approximated fronts.

On a more related analysis to highlight the benefits of the ACR with respect to the B version, Fig. 8 shows the number of epochs in which the HV value of both the B versions and the ACR versions reach a higher HV value. The two subfigures clearly indicates that our proposal better tracks the changing Pareto fronts across the epochs. Of special importance are the impact in DMOCell, which is the best performing algorithm in the

Table 7
Comparison based on the average HV value over all the epochs for the different algorithms for the LL, LM and LH instances.

		LL		LM		LH	
		DNSGA-II	DMOCell	DNSGA-II	DMOCell	DNSGA-II	DMOCell
NRC		$0.1101_{\pm 0.0113}$	$0.1197_{\pm 0.0126}$	$0.1116_{\pm 0.0108}$	$0.1205_{\pm 0.0123}$	$0.1140_{\pm 0.0116}$	$0.1216_{\pm 0.0123}$
A	$r_p = 0.05$	$0.1169_{\pm 0.0143}$	$0.1406_{\pm 0.0151}$	$0.1180_{\pm 0.0147}$	$0.1402_{\pm 0.0145}$	$0.1197_{\pm 0.0144}$	$0.1424_{\pm 0.0144}$
	$r_p = 0.25$	$0.1143_{\pm 0.0121}$	$0.1261_{\pm 0.0124}$	$0.1154_{\pm 0.0134}$	$0.1267_{\pm 0.0132}$	$0.1161_{\pm 0.0138}$	$0.1278_{\pm 0.0124}$
	$r_p = 0.50$	$0.1153_{\pm 0.0135}$	$0.1288_{\pm 0.0141}$	$0.1171_{\pm 0.0138}$	$0.1277_{\pm 0.0129}$	$0.1181_{\pm 0.0132}$	$0.1287_{\pm 0.0126}$
	$r_p = 0.75$	$0.1148_{\pm 0.0122}$	$0.1248_{\pm 0.0118}$	$0.1175_{\pm 0.0135}$	$0.1271_{\pm 0.0129}$	$0.1183_{\pm 0.0135}$	$0.1281_{\pm 0.0132}$
	$r_p = 0.95$	$0.1155_{\pm 0.0131}$	$0.1263_{\pm 0.0128}$	$0.1166_{\pm 0.0127}$	$0.1262_{\pm 0.0131}$	$0.1186_{\pm 0.0134}$	$0.1297_{\pm 0.0127}$
B	$r_p = 0.05$	$0.2433_{\pm 0.0350}$	$0.2733_{\pm 0.0354}$	$0.2454_{\pm 0.0321}$	$0.2707_{\pm 0.0321}$	$0.2485_{\pm 0.0338}$	$0.2764_{\pm 0.0318}$
	$r_p = 0.25$	$0.2677_{\pm 0.0330}$	$0.3003_{\pm 0.0325}$	$0.2708_{\pm 0.0365}$	$0.3028_{\pm 0.0376}$	$0.2747_{\pm 0.0319}$	$0.3070_{\pm 0.0307}$
	$r_p = 0.50$	$0.2786_{\pm 0.0335}$	$0.3204_{\pm 0.0388}$	$0.2820_{\pm 0.0356}$	$0.3244_{\pm 0.0354}$	$0.2844_{\pm 0.0331}$	$0.3272_{\pm 0.0332}$
	$r_p = 0.75$	$0.2854_{\pm 0.0342}$	$0.3375_{\pm 0.0363}$	$0.2863_{\pm 0.0341}$	$0.3409_{\pm 0.0383}$	$0.2926_{\pm 0.0336}$	$0.3473_{\pm 0.0334}$
	$r_p = 0.95$	$0.2879_{\pm 0.0331}$	$0.3504_{\pm 0.0377}$	$0.2901_{\pm 0.0364}$	$0.3522_{\pm 0.0381}$	$0.2972_{\pm 0.0349}$	$0.3636_{\pm 0.0369}$
ACR		$0.2941_{\pm 0.0354}$	$0.3566_{\pm 0.0365}$	$0.2960_{\pm 0.0305}$	$0.3612_{\pm 0.0344}$	$0.3020_{\pm 0.0292}$	$0.3698_{\pm 0.0327}$
ACR^{best}		$0.3581_{\pm 0.03825}$		$0.3612_{\pm 0.04175}$		$0.3718_{\pm 0.03735}$	

Table 8
Comparison based on the average HV value over all the epochs for the different algorithms for the ML, MM and MH instances.

		ML		MM		MH	
		DNSGA-II	DMOCell	DNSGA-II	DMOCell	DNSGA-II	DMOCell
NRC		$0.0781_{\pm 0.0152}$	$0.1001_{\pm 0.0123}$	$0.0760_{\pm 0.0136}$	$0.0981_{\pm 0.0106}$	$0.0771_{\pm 0.0130}$	$0.0991_{\pm 0.0112}$
A	$r_p = 0.05$	$0.1265_{\pm 0.0135}$	$0.1567_{\pm 0.0139}$	$0.1284_{\pm 0.0131}$	$0.1591_{\pm 0.0143}$	$0.1289_{\pm 0.0135}$	$0.1619_{\pm 0.0145}$
	$r_p = 0.25$	$0.1262_{\pm 0.0122}$	$0.1406_{\pm 0.0110}$	$0.1261_{\pm 0.0112}$	$0.1418_{\pm 0.0115}$	$0.1274_{\pm 0.0127}$	$0.1422_{\pm 0.0117}$
	$r_p = 0.50$	$0.1274_{\pm 0.0125}$	$0.1412_{\pm 0.0109}$	$0.1268_{\pm 0.0124}$	$0.1423_{\pm 0.0102}$	$0.1283_{\pm 0.0139}$	$0.1431_{\pm 0.0120}$
	$r_p = 0.75$	$0.1275_{\pm 0.0114}$	$0.1413_{\pm 0.0114}$	$0.1280_{\pm 0.0122}$	$0.1426_{\pm 0.0109}$	$0.1285_{\pm 0.0128}$	$0.1438_{\pm 0.0122}$
	$r_p = 0.95$	$0.1282_{\pm 0.0125}$	$0.1414_{\pm 0.0109}$	$0.1283_{\pm 0.0115}$	$0.1419_{\pm 0.0114}$	$0.1301_{\pm 0.0135}$	$0.1436_{\pm 0.0118}$
B	$r_p = 0.05$	$0.2736_{\pm 0.0362}$	$0.3123_{\pm 0.0344}$	$0.2727_{\pm 0.0313}$	$0.3141_{\pm 0.0329}$	$0.2756_{\pm 0.0362}$	$0.3159_{\pm 0.0338}$
	$r_p = 0.25$	$0.2990_{\pm 0.0346}$	$0.3428_{\pm 0.0327}$	$0.3002_{\pm 0.0341}$	$0.3437_{\pm 0.0307}$	$0.3029_{\pm 0.0352}$	$0.3482_{\pm 0.0320}$
	$r_p = 0.50$	$0.3103_{\pm 0.0378}$	$0.3700_{\pm 0.0343}$	$0.3096_{\pm 0.0359}$	$0.3698_{\pm 0.0293}$	$0.3126_{\pm 0.0349}$	$0.3738_{\pm 0.0335}$
	$r_p = 0.75$	$0.3165_{\pm 0.0328}$	$0.3952_{\pm 0.0378}$	$0.3163_{\pm 0.0346}$	$0.3936_{\pm 0.0334}$	$0.3197_{\pm 0.0380}$	$0.3973_{\pm 0.0335}$
	$r_p = 0.95$	$0.3189_{\pm 0.0320}$	$0.4122_{\pm 0.0383}$	$0.3219_{\pm 0.0353}$	$0.4104_{\pm 0.0325}$	$0.3243_{\pm 0.0350}$	$0.4153_{\pm 0.0373}$
ACR		$0.3237_{\pm 0.0349}$	$0.4172_{\pm 0.0335}$	$0.3237_{\pm 0.0328}$	$0.4164_{\pm 0.0350}$	$0.3246_{\pm 0.0337}$	$0.4202_{\pm 0.0357}$
ACR^{best}		$0.4172_{\pm 0.03990}$		$0.4164_{\pm 0.0335}$		$0.4202_{\pm 0.03639}$	

Table 9
Comparison based on the average HV value over all the epochs for the different algorithms for the HL, HM and HH instances.

		HL		HM		HH	
		DNSGA-II	DMOCell	DNSGA-II	DMOCell	DNSGA-II	DMOCell
NRC		$0.0185_{\pm 0.0136}$	$0.0482_{\pm 0.0123}$	$0.0161_{\pm 0.0115}$	$0.0466_{\pm 0.0117}$	$0.0141_{\pm 0.0131}$	$0.0433_{\pm 0.0134}$
A	$r_p = 0.05$	$0.1183_{\pm 0.0147}$	$0.1552_{\pm 0.0143}$	$0.1223_{\pm 0.0164}$	$0.1561_{\pm 0.0179}$	$0.1199_{\pm 0.0146}$	$0.1539_{\pm 0.0166}$
	$r_p = 0.25$	$0.1177_{\pm 0.0139}$	$0.1383_{\pm 0.0140}$	$0.1253_{\pm 0.0136}$	$0.1377_{\pm 0.0162}$	$0.1201_{\pm 0.0141}$	$0.1359_{\pm 0.0150}$
	$r_p = 0.50$	$0.1194_{\pm 0.0135}$	$0.1354_{\pm 0.0137}$	$0.1253_{\pm 0.0150}$	$0.1386_{\pm 0.0159}$	$0.1215_{\pm 0.0132}$	$0.1366_{\pm 0.0154}$
	$r_p = 0.75$	$0.1203_{\pm 0.0148}$	$0.1358_{\pm 0.0140}$	$0.1294_{\pm 0.0151}$	$0.1389_{\pm 0.0157}$	$0.1210_{\pm 0.0132}$	$0.1364_{\pm 0.0150}$
	$r_p = 0.95$	$0.1205_{\pm 0.0143}$	$0.1358_{\pm 0.0138}$	$0.1245_{\pm 0.0158}$	$0.1394_{\pm 0.0166}$	$0.1223_{\pm 0.0160}$	$0.1372_{\pm 0.0146}$
B	$r_p = 0.05$	$0.2552_{\pm 0.0364}$	$0.2979_{\pm 0.0309}$	$0.2606_{\pm 0.0414}$	$0.3057_{\pm 0.0316}$	$0.2554_{\pm 0.0328}$	$0.3020_{\pm 0.0296}$
	$r_p = 0.25$	$0.2785_{\pm 0.0378}$	$0.3270_{\pm 0.0295}$	$0.2835_{\pm 0.0351}$	$0.3339_{\pm 0.0348}$	$0.2782_{\pm 0.0342}$	$0.3279_{\pm 0.0321}$
	$r_p = 0.50$	$0.2879_{\pm 0.0393}$	$0.3487_{\pm 0.0326}$	$0.2920_{\pm 0.0355}$	$0.3539_{\pm 0.0382}$	$0.2872_{\pm 0.0380}$	$0.3469_{\pm 0.0328}$
	$r_p = 0.75$	$0.2932_{\pm 0.0369}$	$0.3712_{\pm 0.0374}$	$0.2986_{\pm 0.0371}$	$0.3775_{\pm 0.0416}$	$0.2935_{\pm 0.0315}$	$0.3706_{\pm 0.0378}$
	$r_p = 0.95$	$0.2968_{\pm 0.0362}$	$0.3869_{\pm 0.0408}$	$0.3026_{\pm 0.0388}$	$0.3922_{\pm 0.0453}$	$0.2972_{\pm 0.0354}$	$0.3852_{\pm 0.0402}$
ACR		$0.3014_{\pm 0.0342}$	$0.3887_{\pm 0.0300}$	$0.3075_{\pm 0.0373}$	$0.3955_{\pm 0.0343}$	$0.3005_{\pm 0.0296}$	$0.3879_{\pm 0.0310}$
ACR^{best}		$0.3902_{\pm 0.04199}$		$0.3955_{\pm 0.04614}$		$0.3879_{\pm 0.04150}$	

experimentation conducted. This algorithm is able to fully profit from the genetic material the ACR operator passes on to the next epoch, approximating the Pareto fronts accurately.

6. Conclusions and future work

This work has approached the Cell-Switch Off problem in ultra-dense 5G networks from a dynamic evolutionary multi-objective optimization perspective. By considering a set of epochs between which the traffic demand changes, but remains constant within each epoch, the optimization algorithms have been endowed with mechanisms to react to these changes in order to better track the newly induced Pareto front. To improve this tracking, a new restart operator based on exploiting the continuous mobility of the users within a cellular network has been devised. This operator, called Adjacent Cell Restart (ACR), can be easily included in any dynamic MOEA. An extensive and systematic experimentation has been conducted to fully evaluate the impact of the ACR operator in the search of two algorithms: DNSGA-II and DMOCell. The results over nine different UDN scenarios have shown that ACR clearly aids the search of these two algorithms towards better approximations of the Pareto fronts of the considered epochs.

As future research, this work clearly opens promising lines to be further explored. First of all, the evaluation of the ACR operator in other existing MOEAs and additional problem instances

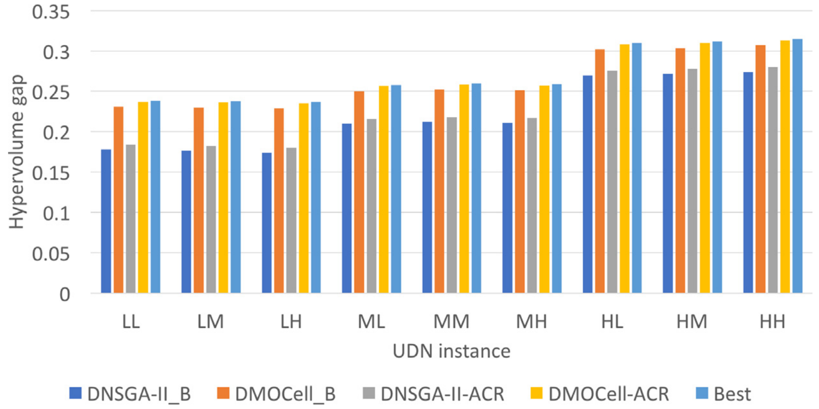
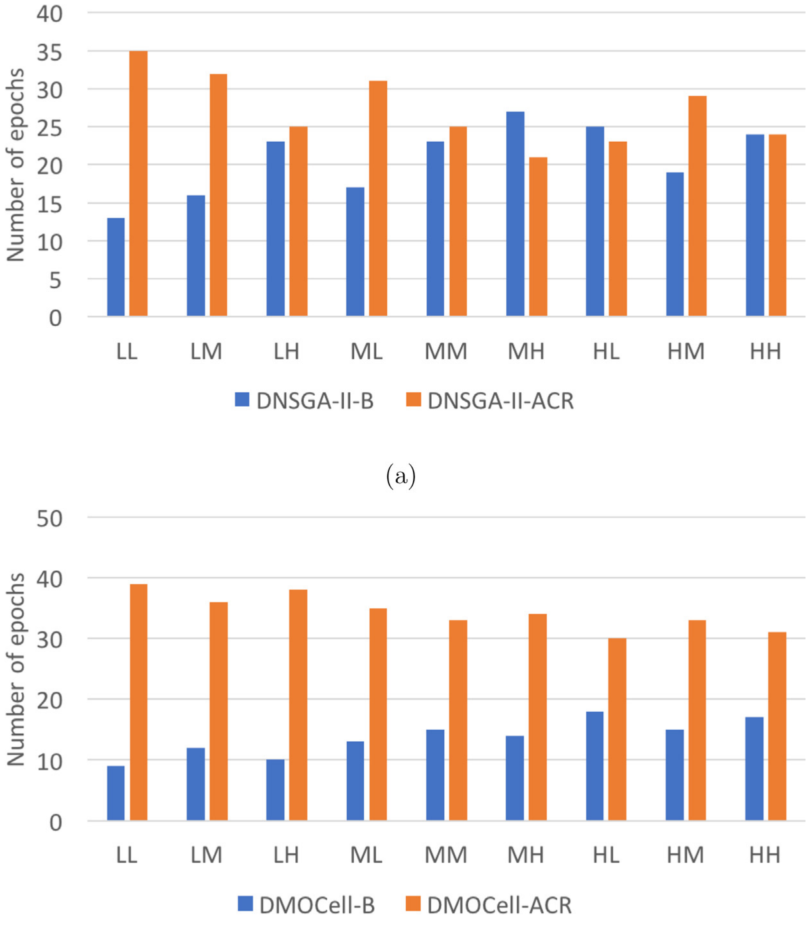


Fig. 7. Average HV gap of the best performing dynamic MOEAs (B version and ACR-enabled) with respect to the NRC version, grouped into the nine scenarios addressed with increasing densification.



(a)

(b)

Fig. 8. Number of winner epochs out of the 48 preconfigured, for the B and ACR versions of (a) DNSGA-II and (b) DMOCell. The column values are grouped into the nine scenarios addressed.

will identify target scenarios in which it may perform better. With respect to the ACR design, it can be extended to exploit not only the spatial continuity of the users, but also to purely add diversity to the search of the algorithms, in a hybrid-like approach with the B versions, which has also shown to perform well. Both memory and prediction based enhanced mechanisms for reacting to change are of great interest. Finally, a further characterization of the impact of the ACR with different settings of the dynamic MOEA (different crossover and mutation rates, different population sizes, etc.) also deserves attention.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.future.2019.10.005.

Acknowledgments

This work has been supported by the TIN2016-75097-P and RTI2018-102002-A-I00 projects of Ministerio de Ciencia, Innovación y Universidades (Spain), and the B-TIC-402-UGR18 project of the Consejería de Economía, Conocimiento, Empresas y Universidad (Junta de Andalucía, Spain). The authors thankfully acknowledges the computer resources, technical expertise and assistance provided by the SCBI (Supercomputing and Bioinformatics) center of the University of Málaga.

References

[1] Ericsson, Ericsson mobility report, 2018, https://www.ericsson.com/en/mobility-report/reports/q4-update-2018. (Accessed June 8, 2019).

[2] Cisco, Global mobile data traffic forecast update, 2017–2022 white paper, 2019, https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-738429.html. (Accessed June 8, 2019).

[3] J.G. Andrews, S. Buzzi, W. Choi, S.V. Hanly, A. Lozano, A.C.K. Soong, J.C. Zhang, What will 5G be?, IEEE J. Sel. Areas Commun. 32 (6) (2014) 1065–1082, http://dx.doi.org/10.1109/JSAC.2014.2328098.

[4] A. Bohli, R. Bouallegue, How to meet increased capacities by future green 5G networks: A survey, IEEE Access 7 (2019) 42220–42237, http://dx.doi.org/10.1109/ACCESS.2019.2907284.

[5] D. Lopez-Perez, M. Ding, H. Claussen, A.H. Jafari, Towards 1 Gbps/UE in cellular systems: Understanding ultra-dense small cell deployments, IEEE Commun. Surv. Tutor. 17 (4) (2015) 2078–2101, http://dx.doi.org/10.1109/COMST.2015.2439636, arXiv:arXiv:1503.03912v1.

[6] X. Ge, S. Tu, G. Mao, C.-X. Wang, T. Han, 5G ultra-dense cellular networks, IEEE Wirel. Commun. 23 (1) (2016) 72–79, http://dx.doi.org/10.1109/MWC.2016.7422408, arXiv:1512.03143.

[7] M. Kamel, W. Hamouda, A. Youssef, Ultra-dense networks: A survey, IEEE Commun. Surv. Tutor. 18 (4) (2016) 2522–2545, http://dx.doi.org/10.1109/COMST.2016.2571730, URL http://ieeexplore.ieee.org/document/7476821/.

[8] M. Yao, M.M. Sohul, X. Ma, V. Marojevic, J.H. Reed, Sustainable green networking: exploiting degrees of freedom towards energy-efficient 5g systems, Wirel. Netw. 25 (3) (2019) 951–960, http://dx.doi.org/10.1007/s11276-017-1626-7.

[9] P. Gandotra, R.K. Jha, S. Jain, Green communication in next generation cellular networks: A survey, IEEE Access 5 (2017) 11727–11758, http://dx.doi.org/10.1109/ACCESS.2017.2711784, URL http://ieeexplore.ieee.org/document/7939957/.

[10] 3GPP, Small Cell Enhancements for E-UTRA and E-UTRAN—Physical Layer Aspects, Tech. rep., 3rd Generation Partnership Project (3GPP), 2014, URL http://www.3gpp.org/ftp/Specs/html-info/36872.htm.

[11] M. Feng, S. Mao, T. Jiang, Base station on-off switching in 5G wireless networks: approaches and challenges, IEEE Wirel. Commun. 24 (4) (2017) 46–54, http://dx.doi.org/10.1109/MWC.2017.1600353.

[12] T. Beitelmal, S. Szyszkowicz, G. David González, H. Yanikomeroglu, Sector and site switch-off regular patterns for energy saving in cellular networks, IEEE Trans. Wireless Commun. 17 (5) (2018) 2932–2945, http://dx.doi.org/10.1109/TWC.2018.2804397.

[13] D. Gonzalez G., J. Hamalainen, H. Yanikomeroglu, M. Garcia-Lozano, G. Senarath, A novel multiobjective cell switch-off framework for cellular networks, IEEE Access 4 (2016) 7883–7898, http://dx.doi.org/10.1109/ACCESS.2016.2625743, arXiv:1506.05595.

[14] W. Lai, C.-S. Shieh, C.-S. Ho, Y.-R. Chen, A clustering-based energy saving scheme for dense small cell networks, IEEE Access 7 (2019) 2880–2893, http://dx.doi.org/10.1109/ACCESS.2018.2886274.

[15] J. Li, H. Wang, X. Wang, Z. Li, Optimized sleep strategy based on clustering in dense heterogeneous networks, EURASIP J. Wirel. Commun. Netw. 2018 (1) (2018) http://dx.doi.org/10.1186/s13638-018-1311-2.

[16] A. Hajijamali Arani, M.J. Omidi, A. Mehbodniya, F. Adachi, Minimizing base stations' on/off switchings in self-organizing heterogeneous networks: a distributed satisfactory framework, IEEE Access 5 (2017) 26267–26278, http://dx.doi.org/10.1109/ACCESS.2017.2777914.

[17] M. Dolfi, C. Cavdar, S. Morosi, P. Piunti, J. Zander, E. Del Re, On the trade-off between energy saving and number of switchings in green cellular networks, Trans. Emerg. Telecommun. Technol. 28 (11) (2017) e3193, http://dx.doi.org/10.1002/ett.3193.

[18] Q.-N. Le-The, T. Beitelmal, F. Lagum, S.S. Szyszkowicz, H. Yanikomeroglu, Cell switch-off algorithms for spatially irregular base station deployments, IEEE Wirel. Commun. Lett. 6 (3) (2017) 354–357, http://dx.doi.org/10.1109/LWC.2017.2690677.

[19] F. Lagum, Q.-N. Le-The, T. Beitelmal, S.S. Szyszkowicz, H. Yanikomeroglu, Cell switch-off for networks deployed with variable spatial regularity, IEEE Wirel. Commun. Lett. 6 (2) (2017) 234–237, http://dx.doi.org/10.1109/LWC.2017.2665472.

[20] A. Salem, S. El-Rabaie, M. Shokair, Energy efficient ultra-dense networks (UDNs) based on joint optimisation evolutionary algorithm, IET Commun. 13 (1) (2019) 99–107.

[21] D. González González, E. Mutafungwa, B. Haile, J. Hämäläinen, H. Poveda, A planning and optimization framework for ultra dense cellular deployments, Mob. Inf. Syst. 2017 (2017) 1–17, http://dx.doi.org/10.1155/2017/9242058.

[22] F. Luna, R. Luque-Baena, J. Martí nez, J. Valenzuela-Valdés, P. Padilla, Addressing the 5G cell switch-off problem with a multi-objective cellular genetic algorithm, in: IEEE 5G World Forum, 5GWF 2018 - Conference Proceedings, 2018, pp. 422–426, http://dx.doi.org/10.1109/5GWF.2018.8517066.

[23] C.A. Coello Coello, G.B. Lamont, D.A. Van Veldhuizen, Evolutionary algorithms for solving multi-objective problems, second ed., Springer, New York, 2007.

[24] S. Jiang, S. Yang, Evolutionary dynamic multiobjective optimization: Benchmarks and algorithm comparisons, IEEE Trans. Cybern. 47 (1) (2017) 198–211, http://dx.doi.org/10.1109/TCYB.2015.2510698.

[25] Y. Jin, J. Branke, Evolutionary optimization in uncertain environments — A survey, IEEE Trans. Evol. Comput. 9 (3) (2005) 303–317, http://dx.doi.org/10.1109/TEVC.2005.846356.

[26] T.T. Nguyen, S. Yang, J. Branke, Evolutionary dynamic optimization: A survey of the state of the art, Swarm Evol. Comput. 6 (2012) 1–24, http://dx.doi.org/10.1016/j.swevo.2012.05.001.

[27] Z. Liang, S. Zheng, Z. Zhu, S. Yang, Hybrid of memory and prediction strategies for dynamic multiobjective optimization, Inform. Sci. 485 (2019) 200–218, http://dx.doi.org/10.1016/j.ins.2019.01.066.

[28] K. Deb, U.B. Rao N., S. Karthik, Dynamic multi-objective optimization and decision-making using modified NSGA-II: A case study on hydro-thermal power scheduling, in: Evolutionary Multi-Criterion Optimization, Springer Berlin Heidelberg, 2007, pp. 803–817.

[29] P. Chandhar, S. Sekhar Das, Multi-objective framework for dynamic optimization of ofdma cellular systems, IEEE Access 4 (2016) 1889–1914, http://dx.doi.org/10.1109/ACCESS.2016.2551640, URL http://ieeexplore.ieee.org/document/7448822/.

[30] L. Aissaoui Ferhi, K. Sethom, F. Choubani, R. Bouallegue, Multiobjective self-optimization of the cellular architecture for green 5G networks, Trans. Emerg. Telecommun. Technol. 29 (10) (2018) e3478, http://dx.doi.org/10.1002/ett.3478.

[31] N. Piovesan, A. Fernandez Gambin, M. Miozzo, M. Rossi, P. Dini, Energy sustainable paradigms and methods for future mobile networks: A survey, Comput. Commun. 119 (December 2017) (2018) 101–117, http://dx.doi.org/10.1016/j.comcom.2018.01.005.

[32] E. Hyytia, P. Lassila, J. Virtamo, Spatial node distribution of the random waypoint mobility model with applications, IEEE Trans. Mob. Comput. 5 (6) (2006) 680–694, http://dx.doi.org/10.1109/TMC.2006.86.

[33] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.

[34] A.J. Nebro, J.J. Durillo, F. Luna, B. Dorronsoro, E. Alba, Mocell: A cellular genetic algorithm for multiobjective optimization, Int. J. Intell. Syst. 24 (7) (2009) 723–725.

[35] C. Liu, B. Natarajan, H. Xia, Small cell base station sleep strategies for energy efficiency, IEEE Trans. Veh. Technol. 65 (3) (2016) 1652–1661, http://dx.doi.org/10.1109/TVT.2015.2413382.

[36] F. Luna, E. Alba, Parallel multiobjective evolutionary algorithms, in: K. J., P. W. (Eds.), Handbook of Computational Intelligence, Springer, 2015.

[37] E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach, IEEE Trans. Evol. Comput. 3 (4) (1999) 257–271.

Francisco Luna received his degree in Engineering and PhD in Computer Science in 2002 and 2008, respectively, from the University of Málaga, Spain, where he was research assistant until 2012. In 2012, he joined Universidad Carlos III of Madrid with a postdoc position, and in 2013 the Universidad de Extremadura as Assistant Professor. Since 2015 he is an Associate Professor at the University of Málaga.

His current research interests include the design and implementation of parallel and multi-objective meta-heuristics, and their application to solve complex problems arising in several domains, including telecommunications, finance, and structural design.

Pablo Helio Zapata Cano (1996, Spain) received the Bachelor Engineering degree in Telecommunications Engineering from the school of Informatics and Telecommunications Engineering from the Universidad de Granada (UGR) in 2019.

Since 2018, he has been collaborating with the "Smart wireless applications and technologies group" and since 2019 he is a young researcher by the Universidad de Málaga (UMA).

In 2019, he presented his bachelor thesis in the topic "Modeling and optimization for the Cell Switch Off Problem in Ultradense 5G Networks". His research interests include optimization techniques applied to 5G networks, green communications and heterogeneous networks.

Juan C. González-Macías received his degree in Telecommunication Engineering in 1992, from the Polytechnic University of Madrid, Spain. He joined Universidad of Extremadura as Contracted Teacher from 1993 and as Assistant Professor from 1998. Since 2002 he is an Associate Professor at the University of Extremadura. His current research interests include the Signal processing and multidisciplinary applications.

Juan F. Valenzuela-Valdés was born in Marbella, Spain. He received his degree in telecommunications engineering from the Universidad de Malaga, Spain, in 2003 and his Ph.D. from Universidad Polit cnica de Cartagena, Spain, in May 2008. In 2004, he joined the Department of Information Technologies and Communications, Universidad Polité cnica de Cartagena. In 2007, he joined EMITE Ing. as head of research. In 2011, he joined Universidad de Extremadura, and in 2015, he joined Universidad de Granada where he is currently an associate professor. His current research areas cover wireless communications and efficiency in wireless sensor networks. He has also been awarded several prizes, including a national prize to the Best Ph.D in Mobile Communications by Vodafone and the i-patents award by Spanish Autonomous Region of Murcia for innovation and technology transfer excellence. He was cofounder of Emite Ing, a spin-off company. He also holds several national and international patents. His publication record is composed of more than 80 publications, including 40 JCR indexed articles, more than 30 contributions in international conferences and 7 book chapter.