

# A Framework for Dynamic Network Architecture and Topology Optimization

Alireza Shams Shafigh, *Student Member, IEEE*, Beatriz Lorenzo, *Member, IEEE*, Savo Glisic, *Senior Member, IEEE*, Jordi Pérez-Romero, *Member, IEEE*, Luiz A. DaSilva, *Senior Member, IEEE*, Allen B. MacKenzie, *Senior Member, IEEE*, and Juha Rönig

**Abstract**—A new paradigm in wireless network access is presented and analyzed. In this concept, certain classes of wireless terminals can be turned temporarily into an access point (AP) anytime while connected to the Internet. This creates a dynamic network architecture (DNA) since the number and location of these APs vary in time. In this paper, we present a framework to optimize different aspects of this architecture. First, the dynamic AP association problem is addressed with the aim to optimize the network by choosing the most convenient APs to provide the quality-of-service (QoS) levels demanded by the users with the minimum cost. Then, an economic model is developed to compensate the users for serving as APs and, thus, augmenting the network resources. The users' security investment is also taken into account in the AP selection. A preclustering process of the DNA is proposed to keep the optimization process feasible in a high dense network. To dynamically reconfigure the optimum topology and adjust it to the traffic variations, a new specific encoding of genetic algorithm (GA) is presented. Numerical results show that GA can provide the optimum topology up to two orders of magnitude faster than exhaustive search for network clusters, and the improvement significantly increases with the cluster size.

**Index Terms**—Access point selection, economic model, network optimization, quality of service (QoS), security investment, topology reconfiguration, traffic dynamics.

## I. INTRODUCTION

IN THE last decade, the mixture of wired and wireless connectivity has transformed the way of accessing the Internet [1]. This heterogeneous connectivity promises to provide

a high level of ubiquitous broadband access. As a result, we are witnessing the emergence of wireless hotspots characterized by the high-density deployment of WLAN access points (APs) [2]. An important feature of the high-density deployment is that users can find multiple APs in its vicinity, from the same or different service providers. Due to the limited number of channels, multiple APs may operate over the same channels. Thus, the effective management of these APs to optimize the users' throughput becomes an important challenge.

To add robustness to the failures of links and/or APs in a wireless mesh network, [3] proposes the overprovisioning of APs in the network. They assume that the APs are connected to the gateway node to access the Internet. However, redundant APs may increase the operational cost and degrade the throughput due to increasing interference. One solution to this problem is to activate only the necessary APs to provide the required connectivity while maximizing the throughput. In this paper, we study further in detail this problem. In particular, as the first step, we proposed a new solution where redundant access points are not static nor occasionally inactive as, for example, small cell concept [4], which is an economically inefficient solution. Instead, a new paradigm in wireless network access is considered where certain classes of wireless terminals (PCs or smartphones) can be turned into an access point anytime while connected to the Internet. Smartphones are currently designed to have such features [5]. This creates a dynamic network architecture (DNA) since the number and location of these access points vary in time. If their access to the Internet is wireless, these new potential access points may be even mobile and change their positions. The aim of this paper is to present a DNA optimization framework to optimize different aspects of this new architecture. First, we optimize the network by choosing the most convenient set of available APs to provide the quality-of-service (QoS) levels demanded by the users. The utility function considered includes the throughput, delay, power consumption, and the cost incurred for having a certain number of APs. To exploit the soft capacity provided by the DNA, an economic model is developed to award the users by adding a credit (negative price) to their bills while acting as APs for other users in their vicinity. This serves as an incentive for the users to efficiently use the network resources. Different options for the pricing mechanism are presented for wired and wireless Internet backhaul. Monopoly market is considered where a single operator controls the market price. Users follow the operator's price control and select the AP that optimizes their utility. This framework brings many opportunities for further work on the microeconomics of DNA. As the change in the terminal's role (from user to AP) can make the

Manuscript received October 25, 2013; revised August 06, 2014 and October 07, 2014; accepted December 04, 2014; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor S. Ramasubramanian. This work was supported by EcoMoCo/Tekes, ADTECH/Huawei, the FP7 NEWCOM# project under Grant No. 318306, Infotech Oulu, Science Foundation Ireland under Grant No. 10/IN.1/13007, and Tactica/MINECO, Spain. This material is based upon work supported by the National Science Foundation under Grant No. 1147790.

A. S. Shafigh and S. Glisic are with the Department of Communications Engineering, University of Oulu, 90550 Oulu, Finland (e-mail: alireza.shamshafigh@ee.oulu.fi; savo.glisic@ee.oulu.fi).

B. Lorenzo was with the University of Oulu, 90550 Oulu, Finland. She is now with AtlantTIC, University of Vigo, 36310 Vigo, Spain (e-mail: blorenzo@gti.uvigo.es).

J. Pérez-Romero is with the Department of Signal Theory and Communications, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain (e-mail: jorperez@tsc.upc.edu).

L. A. DaSilva and A. B. MacKenzie are with the Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24061 USA (e-mail: ldsilva@vt.edu; mackenzie@vt.edu).

J. Rönig is with the Department of Computer Science and Engineering, University of Oulu, 90550 Oulu, Finland (e-mail: jjr@ee.oulu.fi).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNET.2014.2383437

system prone to eavesdropping, the user's requirements in terms of security are also considered in the selection of the AP.

As the traffic in the network changes due to mobility, new/ended calls (sessions), or new/ended service of APs, a two-level access admission control (2L-AAC) scheme is presented to accept/postpone the requests from new users and APs to join/leave the network and keep the network utility at the required level. In order to keep the optimization process feasible in a high dense network, a preclustering process of the network is proposed. A specifically encoded genetic algorithm is developed to dynamically reconfigure the optimum topology and adjust it to the changes in the network.

Bio-inspired algorithms have been used in [6]–[9] for topology control due to their effectiveness for the solution of NP-hard problems. All other bio-inspired routing protocols, e.g., swarm intelligence based routing in ant colony system models [10], react slowly to the changes in the network. Instead, genetic algorithm (GA) reacts instantaneously to those changes, which makes it a very convenient algorithm for controlling topology reconfiguration.

The contributions presented in this paper can be summarized as follows: 1) a new DNA; 2) DNA optimization process to minimize the number of APs needed to satisfy the QoS requirements from the users and thus minimize the network cost; 3) economic model to award the users for acting as APs and share their remaining prepaid traffic volume for other users' transmissions; 4) pricing model that reflects the security investment and security requirements in the AP selection; 5) a 2L-AAC procedure that provides access for the new users/APs to the network based on the required level of utility; 6) specific encoding of genetic algorithm for dynamic reconfiguration of the optimum topology in order to adjust it to the changes in the traffic.

The rest of the paper is organized as follows. The most related work is discussed in Section II. In Section III, the network model and the notation used throughout the paper is presented. The dynamic network architecture optimization is described in Section IV. Section V proposes a genetic algorithm for topology reconfiguration. The admission control mechanism together with implementation details are also provided in this section. An extensive set of results is given in Section VI. Finally, Section VII concludes the paper.

## II. RELATED WORK

Most existing IEEE 802.11 products select an AP based on the signal strength. Unfortunately, this approach has been demonstrated in [11] to result in poor user experience since it does not take the actual load distribution among APs into account. To address this issue, a number of works propose other alternative metrics for achieving a more reasonable association, such as the packet error rate with the number of associated users [12], the throughput model [13], and the potential bandwidth a user can obtain [14]. Channel selection is another major problem in the area of AP selection. As mentioned in [15], this should not be separated from the load balancing of users among APs. The purpose of assigning an appropriate channel is to minimize the inter-AP interference since APs operating in the same channel share the bandwidth. Currently, a widely used method is Least Congested Channel Search (LCCS) [16], which chooses a channel that has the least possibility to conflict

with others in terms of its radio frequency site survey. As more APs are deployed, some papers [17], [18] try to alleviate the inter-AP interference by adjusting the carrier sensing range. Reference [17] indicates that AP-centric methods, in fact, are unable to detect potential interference out of the AP's sensing range. Thus, it proposes a conflict set coloring model to consider the existence of clients among APs during channel selection. The algorithm in [18] requires collecting the interference relation within the one-hop range of the AP–user link, which is challenging work especially where APs are managed by different operators.

The problem of load balancing in 802.11 WLAN is studied in [19] and [20]. The former proposes an association control to obtain the fairest solution, in terms of max-min fairness, whereas the latter proposes a mechanism to allocate the mobile users to the most appropriate AP, taking into account both user preferences and network context. In addition to AP selection, [21] considers the hotspot problem and aims to ensure high resource availability to combat any unexpected traffic growth in the network. In [22], the joint AP selection and power allocation problem is considered in a multicarrier wireless network with multiple APs and mobile users. The APs operate on nonoverlapping spectrum bands, each of which is divided into parallel channels. Each user's objective is to find a suitable AP, followed by a power optimization.

Interesting overviews on the potentials of game theory in the field of network selection are provided in [23] and [24]. In [25], a noncooperative game is used to model the association process to Wi-Fi-based APs. The cost function that each user aims at minimizing depends on the AP load and the distance the associating device needs to travel to be able to associate to the desired AP. Wi-Fi association is addressed through game-theoretic tools also in [26]–[28]. Those works consider noncooperative game models with the users trying to selfishly minimize a cost function that depends on the current congestion of the Wi-Fi access points.

In [29], an AP selection scheme for 802.11 networks is presented that takes into account the available bandwidth, cost of the connection, and security level of the candidate AP. They show through simulations the impact of the throughput on the AP selection based on the security protocol used by the AP. In this paper, the user security investment is included in the pricing model and the probability that the user will be elected when acting as an AP will depend on its investment.

The economic aspects of access networks have been addressed in [30]–[32]. In [30], the authors studied the economic interaction between the Wi-Fi and WiMAX network providers. In [31], a cell-dimensioning approach, which maximizes the aggregate economic utility in a code division multiple-access (CDMA) cell, was proposed to accommodate the optimal cell coverage according to user preferences. Reference [32] dealt with business dynamics of open Wi-Fi networks. They claim that the evolution of such network depends on the initial coverage, subscription fee and user preference. Instead, our work focuses on the economic aspect of users' willingness to act as AP and thus, contribute to extend the network resources.

The closest paper to our DNA model is [33], in which femtocells are introduced as additional potential Internet gateways to enhance the performance of wireless mesh networks. They focus on the femtocells' unpredictable operating times and

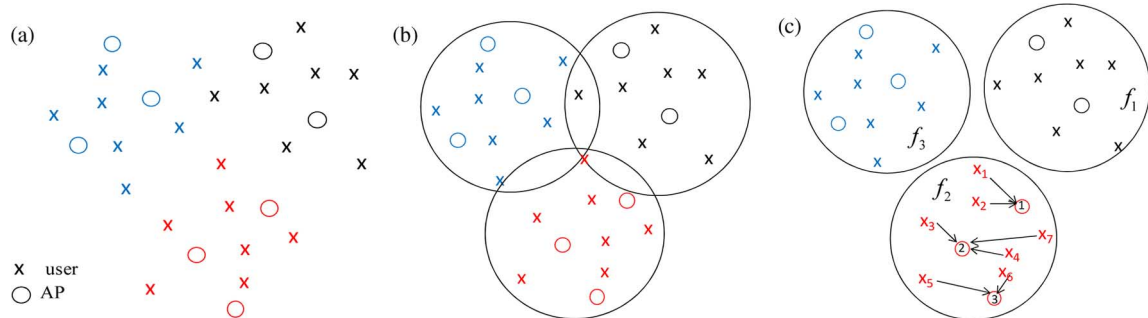


Fig. 1. Dynamic network architecture: (a) possible realization of DNA macro network; (b) clustering DNA macro network; (c) formal cluster separation in DNA network for different frequencies ( $f_k, k = 1, 2, 3$ ) and example of transmission between a cluster of DNA network.

design distributed switching protocols for the mesh routers to switch between different femtocells. Besides, femtocells differ from our DNA in the following aspects: They are user-deployed base stations that bring extra deployment/maintenance cost. They have fixed locations and connect to the cellular network through residential DSL connection. Thus, they rely on the availability of wired connections and infrastructure installations. Although there are few efforts that address the openness of femtocells [34], they mostly require already registered users, so the economic models adopted in those networks are based on different business aspects [35].

Also, femtocells connect to the operator's network over the Internet through a home gateway that includes security functionalities. In our DNA, the terminal (PC or smartphone) is in charge of providing the security level demanded by the users. All additional contributions of the paper, which distinguish our paper from the existing literature, are summarized in Section I.

### III. SYSTEM MODEL

#### A. Network Model

We consider an advanced wireless technology where a subset of terminals can be turned into an access point. With a slight modification of the existing technology, a PC connected to the Internet can serve as an access point. Mobile phones have been recently designed to have such futures [5]. Each terminal independently makes the decision to become an AP, and incentives for such a decision are discussed in Section IV-C. This technology creates the possibility that a number of potential access points can be activated to serve a set of wireless terminals in their vicinity in an optimal way in accordance with some optimization criteria. This results into a DNA.

A high dense network, as shown in Fig. 1(a), is considered. To be able to handle such a dense network and solve efficiently the problems posed in the next sections, the DNA macro network is divided into clusters as shown in Fig. 1(b). The optimization will be solved per cluster as indicated in Fig. 1(c). The intercluster interference is eliminated by using frequency reuse factor as shown in the same figure. Hence, the problems addressed in the sequel will be solved first per DNA cluster, and later on, further comments will be given on how to solve those problems at the level of the macro network. For clarity of presentation, we will refer by DNA to a DNA cluster within the DNA macro network.

We assume that there are  $K$  available potential access points and  $N$  users placed randomly in the DNA. Both, the location of the users and access points change over time. We assume that

the users in a DNA share a given channel to transmit uplink to the different APs. The resource allocation problem consists of allocating to  $N$  users,  $M$  out of  $K$  ( $M \leq K$ ) possible access points in an optimal way.

As the focus of this work is on high dense networks, the channel model considered includes the propagation losses, but not the effects of fading due to the proximity between the users [36]. Thus, if we denote by  $P_i$  the transmission power of user  $i$ , then the received power at AP  $j$  is

$$P_j = h_{ij} P_i \approx d_{ij}^{-\alpha} \cdot P_i \quad (1)$$

where  $h_{ij}$  is the channel gain between user  $i$  and AP  $j$ ,  $d_{ij}$  is the transmission distance between them, and  $\alpha$  is the path-loss factor. The short transmission distance enabled by the availability of additional access points will significantly reduce the impact of fading. The issue of channel defading has been recently discussed in [36]. It was argued that in multihop wireless networks, by reducing the distance between the receiver/transmitter pairs, a point should be reached where the multipath component can be neglected, and the general channel coefficient  $h_{ij}$  that includes the fading can be approximated by the attenuation due to distance only  $\sim d^{-\alpha}$ . The same principle applies to the case considered in this paper where additional access points are chosen on the distance that will enable channel defading. Elaboration of the fading issues as a standard problem has been minimized in order to make space for unconventional issues created by the new architecture presented in this paper. In the general case, instead of distance, the channel state information should be available in the optimization process.

We assume that user  $i$  can successfully connect to AP  $j$  only if the power received at  $j$  exceeds the receiver sensitivity  $\tau$ . As result, the distance between them must satisfy  $d_{ij} \leq (P_i/\tau)^{1/\alpha}$ . For a given distance  $d_{ij}$ , the minimum transmission power for user  $i$  to reach the access point is denoted as  $P_{iu} = \tau \cdot d_{ij}^\alpha$ . If the available power at user  $i$ ,  $P_{ia}$  is less than  $P_{iu}$ , then the connection will not be established. We also define  $r_{iu} = (P_{iu}/\tau)^{1/\alpha}$  as the required communication range of user  $i$ .

In a high dense network, the aim is to keep the transmission power  $P_{iu}$  as low as possible, so that the interference is reduced and higher number of simultaneous transmissions can coexist in the network.

We denote by  $\mathbf{T} = [T_{ij}]$  the network topology matrix with entries  $T_{ij} = 1$  if user  $i$  transmits to AP  $j$ , or zero otherwise.  $T$  provides information of which user is transmitting to which AP for all pairs  $(i, j)$  of user  $i$ ,  $1 \leq i \leq N$ , and AP  $j$ ,  $1 \leq j \leq K$ , and it has dimensions  $N \times K$ .

We assume the following connectivity constraints.

— Each AP can serve one user at a time

$$\mathbf{1} \cdot \mathbf{T} = \mathbf{a}, \quad a_j \leq 1 \quad (2a)$$

where  $\mathbf{1}$  is a  $1 \times N$  vector with the value 1 in every entry.

— Each user will select one AP at a time

$$\mathbf{T} \cdot \mathbf{1} = \mathbf{u}, \quad u_i \leq 1 \quad (2b)$$

where  $\mathbf{1}$  is a  $K \times 1$  vector with the value 1 in every entry.

The topology  $\mathbf{T}$  is a *feasible* topology when the above constraints are satisfied and provides connectivity for all users  $N$  through the available access points  $K$  (i.e.,  $\text{Rank}(\mathbf{T}) = N$ ). The set  $\mathcal{T}$  represents the set of all possible *feasible* topologies  $\mathbf{T}$ , for all possible pairs  $(i, j)$ .

Under the condition that we want to allocate  $K$  access points to all users in the same time-slot ( $N = K$ ), the capacity of an uplink between the user  $i$  and access point  $j$  for certain topology  $\mathbf{T} \in \mathcal{T}$  is

$$c_{ij}(\mathbf{T}) = \rho_{ij} \log \left( 1 + \frac{h_{ij} \cdot P_i}{N_0 + \sum_{k \neq i} h_{kj} \cdot P_k} \right) \quad (3)$$

where it is assumed that all APs work on the same frequency channel and  $\rho_{ij}$  is a binary variable indicating the accessing status of user  $i$ . We have  $\rho_{ij} = 1$  if the user  $i$  can communicate with the AP  $j$ ; otherwise,  $\rho_{ij} = 0$ . That is, by using (1)

$$\rho_{ij} = \begin{cases} 1, & \text{if } d_{ij} \leq r_{iu} \text{ and } P_i = P_{iu} \leq P_{ia} \\ 0, & \text{otherwise} \end{cases} \quad (3a)$$

where  $d_{ij}$  is the distance between the user  $i$  and its access point  $j$ . The interfering cochannel signal at access point  $j$ ,  $d_{kj}^{-\alpha} \cdot P_k$ , is generated by user  $k$ ,  $k \neq i$  while transmitting to its own access point. The background noise power is denoted by  $N_0$ . The overall network capacity is then defined as

$$C = \sum_{i=1}^N c_{ij}(\mathbf{T}). \quad (4)$$

If  $N > K$  or the spatial distribution of potential access points and users is such that none of the possible topologies  $\mathbf{T} \in \mathcal{T}$  can provide satisfactory performance, then some subsets of users might be scheduled in different time slots on the TDMA principle.

For this purpose, we redefine the topology as a block matrix  $\mathbf{T} = [\mathbf{1} \mathbf{T}^2 \mathbf{T} \dots \Delta \mathbf{T}] = [\delta \mathbf{T}]$  where each submatrix is a partial topology  $\delta \mathbf{T}$  per slot  $\delta$ ,  $\delta = 1, \dots, \Delta$ , and  $\Delta$  is the length of the scheduling cycle.  $\delta \mathbf{T}$  provides the information of simultaneous transmissions in slot  $\delta$ . The dimension of  $\mathbf{T}$  is now  $\Delta \times N \times$

$K$ .  $\mathbf{T}$  is a feasible topology if  $\delta \mathbf{T}$  satisfies the connectivity constraints given by (2a) and (2b) for every  $\delta$ ,  $\delta = 1, \dots, \Delta$  and provides connectivity to  $N$  users through  $K$  access points during the scheduling length  $\Delta$  (i.e.,  $\sum_{\delta} \text{Rank}(\delta \mathbf{T}) = N$ ). It is worth noticing that not all APs,  $K$ , need to be used for the topology to be feasible. Actually, we will be mostly interested to find a subset of the available APs that provide the users' QoS requirements.

The scheduling set  $S$  is defined as the set  $S = \{\Delta/\mathbf{T} \in \mathcal{T}, \mathbf{T} = [\delta \mathbf{T}], \delta \mathbf{T} \in \Pi, \delta = 1, 2, \dots, \Delta\}$ , where  $\Pi$  denotes the collision-free set of transmissions. By definition, there is no collision in the transmission between user  $i$  and AP  $j$ , and interference user  $k$  and its AP  $j_k$ , if  $d_{ik} > d_{ij}$  and  $d_{ij_k} > d_{ij}$ .

The topology matrix, for the example presented in Fig. 1(c), is shown at the bottom of the page, where users 1 and 5 transmit on the first slot to access points 1 and 3, respectively. Users 2 and 4 share the second slot transmitting to the APs 1 and 2, respectively. Users 3 and 6 transmit in the third slot to APs 2 and 3, respectively. Finally, user 7 transmits on the fourth slot to AP 2.

The overall network capacity can be obtained as in (4) with  $\mathbf{T} = [\delta \mathbf{T}]$ ,  $\delta = 1, 2, \dots, \Delta$ .

The focus of this work is on uplink transmission, but the same model could be used for downlink as well by considering the AP transmission range, AP transmission power, and signal-to-interference-plus-noise ratio (SINR) received by the users. It is worth noticing that the connectivity constraints for downlink transmission may result into different feasible topologies.

### B. Network Dynamics and Approximation for High Dense Scenarios

The network architecture and thus the topology in DNA will change in time due to new/ended calls or new/ended service of APs. To reconfigure the optimum topology to the traffic changes in a high dense network, some simplifications are needed for practical implementation.

If we denote by  $\lambda_m$  the call arrival rate, the computational time  $T_c$  needed to obtain the new optimum topology after a traffic change should satisfy the following constraint:

$$T_c < 1/\lambda_m. \quad (5)$$

In this way, the new topology can track the network dynamics. To keep the computational complexity under that threshold, the size of the DNA macro network should be scaled down accordingly. For this purpose, we assume that the DNA

$$\mathbf{T} = \begin{matrix} i \setminus AP & 1 & 2 & 3 & & 1 & 2 & 3 & & 1 & 2 & 3 & & 1 & 2 & 3 \\ & & & & \delta=1 & & & & \delta=2 & & & & \delta=3 & & & & \delta=4 \end{matrix} \begin{bmatrix} \left( \begin{array}{ccc} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) \\ \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) \\ \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{array} \right) \\ \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) & \left( \begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right) \end{bmatrix}.$$

macro network is divided into clusters where its size, in terms of  $N$  and  $K$ , is such that constraint (5) holds. The concept of DNA clustering is shown in Fig. 1. More detailed comments on how to optimize different aspects of the DNA macro network will be given in Section IV. Notes on the practical implementation will be provided in Section V.

#### IV. ACQUISITION OF THE OPTIMUM NETWORK ARCHITECTURE

In this section, a number of optimization problems are presented for the DNA paradigm. In the first step, the aim is to develop a basic algorithm to find the optimum topology  $\mathbf{T} \in \mathcal{T}$  in accordance with some utility function. Later on, this algorithm will be used to solve the topology optimization in more sophisticated problems. The QoS requirements will be included as constraints of the optimization problems with the objective to find the minimum required number of APs. Then, an economic model is developed to compensate the users for acting as APs and thus contribute to increase the network resources. Finally, the security requirements are considered.

##### A. Topology Optimization for Fixed Number of APs

We start by considering that the DNA network consists of  $N$  users and  $K$  access points that will be referred to as DNA( $N, K$ ). There are a number of possibilities to allocate all these  $N$  users to  $K$  access points. Each option defines a feasible topology  $\mathbf{T} \in \mathcal{T}$ . The utility function is defined as the sum of the utilities per user  $i$

$$U = \sum_{i=1}^N U_i = \sum_{i=1}^N \frac{c_{ij}(\mathbf{T})}{\Delta \cdot P_i} \quad (6)$$

and includes the link capacity  $c_{ij}(\mathbf{T})$  between user  $i$  and AP  $j$  when transmitting by topology  $\mathbf{T}$  defined by (3), the scheduling length  $\Delta$ , and the power consumption  $P_i$ . All these parameters have impact on the election of the optimum topology. In order to keep the overall transmission power as low as possible, we assume that each user transmits with the minimum transmission power needed to reach the access point  $P_i = P_{iu}$ . We also assume that the users' available power  $P_{ia} \geq P_{iu}$ . Then, the optimum topology is obtained by solving the following optimization problem:

$$\begin{aligned} \mathcal{P}_1 : \underset{\mathbf{T}}{\text{maximize}} \quad & \sum_{i=1}^N \frac{c_{ij}(\mathbf{T})}{\Delta \cdot P_i} \\ \text{subject to} \quad & \mathbf{T} = [\delta \mathbf{T}], \quad \delta = 1, 2, \dots, \Delta \\ & \mathbf{T} \in \mathcal{T}, \quad \Delta \in S \\ & \sum_{\delta} \text{Rank}(\delta \mathbf{T}) = N \\ & P_i = P_{iu}, \quad P_i \leq P_{ia}. \end{aligned} \quad (7)$$

As result, the optimum topology  $\mathbf{T}^*$  that satisfies the previous constraints is obtained where  $\Delta \in S$ . Although this problem is NP-hard [37], the utility considered allows certain simplifications.

The dependency between the utility and power follows the relation  $\log(P)/P$ , so lower power translates into higher utility. At the same time, lower power will imply lower  $\Delta$  as more simultaneous transmissions can coexist in the network. The load

distribution is considered in (7) through the scheduling length  $\Delta$ . As we have assumed that users will transmit one at a time to a particular AP, a higher number of users allocated to the same AP will result into higher  $\Delta$ , and thus lower utility. For these reasons, this optimization will provide as result for optimum topology  $\mathbf{T}^*$  the one that connects the users to its closest available AP. Based on this result, a Minimum Distance Clustering/Scheduling (MDCS) scheme can be used for topology optimization. In MDCS, the users transmit to their closest AP on a cluster basis. Although there are many options to perform the scheduling, this scheme significantly reduces the topology search space. The scheduling between adjacent clusters in DNA can be performed with a temporal offset in terms of slots or spatial offset by allocating different frequencies to each cluster. This will result into different reuse factor  $\Gamma$  for the DNA macro network, DNA( $N, K, \Gamma$ ), as shown in Fig. 1(c) for  $\Gamma = 1/3$ .

In order to provide the incentive for the terminals to serve as access points in a given time period, the network will have to compensate such a service by paying them a certain amount in a normalized currency that will be discussed later. As the cost of having a large number of access points  $K$  may be too high, in the sequel we study the possibility of activating  $M \leq K$  available APs while still satisfying the users' QoS requirements.

##### B. Topology and Architecture Optimization

The optimization problem defined in (7) can be modified to include the cost incurred by having a certain amount of APs. The aim is now to obtain the optimum topology  $\mathbf{T}$  and the number of APs,  $M$  that guarantees the connectivity for all users  $N$  and provides the maximum utility and minimum cost.

Let  $\boldsymbol{\sigma}$  be a  $1 \times K$  vector where each component  $\sigma_j$  is a binary variable indicating the selection of AP  $j$  with entries  $\sigma_j = 1$  if the AP  $j$  is elected; otherwise, entries  $\sigma_j = 0$ . The vector where each component is the opposite to each component of  $\boldsymbol{\sigma}$  is denoted by  $\bar{\boldsymbol{\sigma}}$ . Its transpose is given as  $(\bar{\boldsymbol{\sigma}})^T$ .

The topology and architecture optimization problem is defined as follows:

$$\begin{aligned} \mathcal{P}_2 : \underset{\mathbf{T}, \boldsymbol{\sigma}}{\text{maximize}} \quad & \sum_{i=1}^N \sum_{j=1}^K \sigma_j \left( \frac{c_{ij}(\mathbf{T})}{\Delta \cdot P_i} - \nu \text{cost}_j \right) \\ \text{subject to} \quad & \mathbf{T} = [\delta \mathbf{T}], \quad \delta = 1, 2, \dots, \Delta \\ & \mathbf{T} \in \mathcal{T}, \quad \Delta \in S \\ & \mathbf{T} \cdot (\bar{\boldsymbol{\sigma}})^T = \mathbf{0} \\ & \sum_{\delta} \text{Rank}(\delta \mathbf{T}) = N \\ & P_i = P_{iu}, \quad P_i \leq P_{ia} \end{aligned} \quad (8)$$

where  $\nu$  is a scaling parameter. The first and second constraints state that the topology  $\mathbf{T}$  should be a feasible topology, the third constraint indicates that the users should be allocated just to active APs, the fourth constraint indicates that  $\mathbf{T}$  should provide connectivity for all users  $N$ , and finally, the fifth constraint establishes the power limits.

As result, the feasible set of topologies  $\mathcal{T}$  can be obtained by MDCS scheme for each  $\boldsymbol{\sigma}$ . Then, the optimum  $\mathbf{T}^* \in \mathcal{T}$  and  $\boldsymbol{\sigma}^*$  that maximize the utility for certain cost are jointly chosen. The number of selected APs,  $M$ , that maximizes the utility is obtained as  $M = \sum_{j=1}^K \sigma_j$ . If the  $\text{cost}_j$  is the same for every AP, then the previous optimization will provide the minimum

$M$ . Otherwise, it will lead to configurations that avoid using high-cost APs.

*QoS Requirements:* We consider that the users' QoS requirement is given in terms of throughput by the following constraint:

$$c_{ij}(\mathbf{T})/\Delta \geq \gamma \quad (9)$$

where  $\gamma$  is a constant. The optimization problem defined by (8) can be reformulated to minimize the number of APs,  $M$ , needed to satisfy the QoS requirements. This can be formulated as

$$\begin{aligned} \mathcal{P}_3 : \underset{\mathbf{T}, M}{\text{minimize}} \quad & M = \sum_{j=1}^K \sigma_j \\ \text{subject to} \quad & \mathbf{T} = [\delta \mathbf{T}], \quad \delta = 1, 2, \dots, \Delta \\ & \mathbf{T} \in \mathcal{T}, \quad \Delta \in S \\ & \sum_{\delta} \text{Rank}(\delta \mathbf{T}) = N \\ & \mathbf{T} \cdot (\bar{\boldsymbol{\sigma}})^T = \mathbf{0} \\ & P_i = P_{iu}, \quad P_i \leq P_{ia} \\ & c_{ij}(\mathbf{T})/\Delta \geq \gamma \end{aligned} \quad (10)$$

where the constraints are the same as in (8) plus the QoS constraint given by (9).

As before, if we assume that the cost is the same for every AP, then the previous optimization will implicitly minimize the cost. In the sequel, different options for the cost will be studied together with its impact on the AP selection.

The previous optimization problems are solved assuming that the backhaul capacity will not limit the capacity obtained in the wireless links, and thus, the QoS requirements can be achieved in problem (10). Additional comments on the potential capacity constraints imposed by the backhaul links are given in the sequel.

### C. Economic Model for Resource Harvesting in DNA

As described in the previous sections, in DNA, a terminal can operate either as a user consuming the network resources or as an AP augmenting the network resources. In the former case, the terminal will pay to the network an amount proportional to the resource consumption, while in the latter case, the network will pay to the terminal an amount proportional to its contribution to the overall augmentation of the network resources. To reduce the cost that APs bring to the network, the users will choose the AP that offers the minimum price. For this reason, we need to specify in more details the contract between the terminal and the operator(s). In the sequel, we present different options for these contracts

*T/W(r/q)/I(R/Q) contract.*

In this option, the terminal  $T$ , when acting as a user, has a contract with the wireless operator  $W$ , and separate contract with the Internet operator  $I$ . Both contracts may be based on the rate  $r$  provided by the wireless operator (rate  $R$  provided by Internet operator) or upload traffic volume  $q$  in the wireless connection (traffic volume  $Q$  on the Internet connection).

The pricing mechanism may be designed in such a way that the price is proportional to the real rate  $r(t)$  and time  $T_r$  the channel is used. This is designated as

$$\text{price}(r, T_r) = \alpha_r \int_0^{T_r} r(t) dt \quad (11a)$$

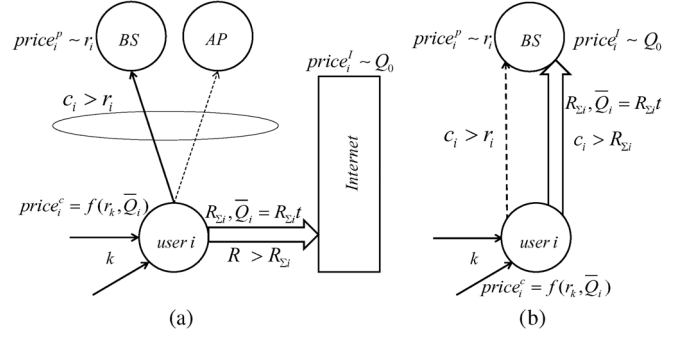


Fig. 2. DNA network model with  $T/W(r)/I(Q)$  contract with (a) wired Internet and (b) wireless Internet.

where  $\alpha_r$  is a proportionality constant. If the operator provides a fixed guaranteed rate  $r_0$ , then

$$\text{price}(r_0, T_r) = \alpha_r \int_0^{T_r} r_0 dt = \alpha_r r_0 T_r. \quad (11b)$$

Instead of rate, the pricing might be based on the maximum upload traffic volume  $q$  during time  $T_q$  resulting in fixed price

$$\text{price}(q, T_q) = \alpha_q \int_0^{T_q} r(t) dt \quad (12a)$$

where  $\alpha_q$  is a proportionality constant, or for the fixed rate

$$\text{price}(q_0, T_q) = \alpha_q \int_0^{T_q} r_0 dt = \alpha_q r_0 T_q. \quad (12b)$$

Similar expressions can be used for the Internet where instead of  $r$  and  $q$  notation,  $R$  and  $Q$  are used. Depending on the pricing mechanism used, a specific contract may have four different options:  $T/W(r)/I(R)$ ,  $T/W(r)/I(Q)$ ,  $T/W(q)/I(R)$ , and  $T/W(q)/I(Q)$ . Moreover, on the Internet segment, there is the possibility that the AP has wired or wireless Internet through the cellular network.

To illustrate the previous discussion, an example for  $T/W(r)/I(Q)$  contract is shown in Fig. 2 including different options for DNA infrastructure for wired and wireless Internet. In particular, in Fig. 2(a), the user can transmit wirelessly through the base station (BS) or AP, and wired to the Internet. The price that terminal  $i$  will pay for the wireless connection,  $\text{price}_i^p$ , depends on its transmission rate  $r_i$ , while the price of the Internet service,  $\text{price}_i^l$ , depends on the prepaid traffic volume,  $Q_0$ . The price that the terminal will charge to any adjacent user  $k$  for the connection is referred to as  $\text{price}_i^e$  and depends on its rate  $r_k$  and the traffic volume  $\bar{Q}_i$  previously used at that terminal  $i$ . We denote by  $R_{ei}(I_{ki}) = \sum_k I_{ki} r_k$  the overall external transmission rate of terminal  $i$  due to the adjacent users  $k$  that transmit to  $i$  when acting as AP. The indicator  $I_{ki} = 1$  if user  $k$  transmits through  $i$ , and zero otherwise. The overall rate on the Internet connection  $R$  is constrained by the transmission rate of terminal  $i$ ,  $R > R_{\Sigma_i} = R_i + R_{ei}$ , where  $R_{\Sigma_i}$  is the aggregated rate of terminal  $i$  and its adjacent users. The rate  $R_i$  will be referred to as the internal rate of the terminal  $i$  on the Internet and  $R_{ei}$  the external transmission rate of the adjacent users. The traffic volume transmitted by  $i$  up to time  $t$  is given by  $\bar{Q}_i = R_{\Sigma_i} t$ . The same notation applies for Fig. 2(b), with the only difference that the Internet connection

is wireless. Thus, the overall rate on the wireless Internet connection is constrained by the link capacity,  $c_i > R_{\Sigma i}$ . Although multiple-input-multiple-output (MIMO) (massive MIMO) technologies can significantly increase  $c_i$ , the new access point in this case cannot still offer capabilities of WLAN. Even so, offloading a number of macrocell users through the new AP will reduce the interference, and so increase the capacity for other potential users in the network. In the example shown in Fig. 2(a), such a constraint does not exist.

In order to analyze the balance on the terminal's account, and thus the price that the terminal will charge to its adjacent users when acting as an AP, we need to average out the terminal operation in time. We will assume that a terminal has a contract of the type  $T/W(r)/I(Q)$ , as shown in Fig. 2, signed with a single operator, which means that the same operator provides both the  $W$  and the  $I$  service.

Let us assume that the user pays to the Internet operator a fixed price  $price_i^I = \chi_0$  for transmitting a maximum traffic volume  $Q_0$  on the Internet connection during period  $T_0$ . We denote by  $p^I$  the probability of having the Internet connection available at a given moment. The probability that the user transmits its own traffic on the Internet connection is given by  $p_i^I$ , and the probability of using instead the wireless connection is  $p_i^W$ .

Then, the average traffic volume sent by terminal  $i$  on the Internet connection until time  $t$  is given by

$$\bar{Q}_i(t) = p^I (p_i^I R_i t + (1 - p_i^I) R_{ei} t) \quad (13a)$$

where  $R_i$  is the transmission rate of terminal  $i$  and  $R_{ei}$  is its average external rate. To define this parameter, we first define the probability  $p_{ki}$  that  $i$  is in the transmission range of user  $k$ . This probability can be expressed as  $p_{ki} = A_k/A_c$ , where  $A_k$  and  $A_c$  are the transmission area of user  $k$  and area of the DNA cluster, respectively. Now, the average overall external rate at terminal  $i$  can be presented as

$$R_{ei} = r \sum_{k=0}^N k (e_i p_{ki})^k \cdot (1 - e_i p_{ki})^{N-k} \quad (13b)$$

where  $r = \bar{r}_k$  is the average user rate on the wireless connection and  $e_i$  is the probability that terminal  $i$  is elected as AP. The probability  $e_i$  is supposed to be inversely proportional to the price that the terminal  $i$  will charge to its adjacent users for the connection that is referred to as  $price_i^c$ . This price depends on the fixed price  $\chi_0$  paid for the Internet and remaining traffic volume available at  $i$

$$price_i^c = \frac{\chi_0}{1 + Q_0 - \bar{Q}_i(t)}, Q_0 \geq \bar{Q}_i(t) \quad (14)$$

where  $\bar{Q}_i(t)$  is the cumulative traffic transmitted until time  $t$ . As the available traffic volume at terminal  $i$  decreases, the higher will be the price it will charge to its adjacent users for the connection. Then, by modeling  $e_i = \alpha_e / price_i^c$  where  $\alpha_e$  is a proportionality constant, so that  $0 \leq e_i \leq 1$ , and using it in (13b),  $\bar{Q}_i(t)$  can be evaluated. For simplicity, we assume that only one user can access terminal  $i$ , thus (13a) can be rewritten for  $R_{ei} = r$  as

$$\begin{aligned} \bar{Q}_i(t) &= p^I (p_i^I R_i t + (1 - p_i^I) r t \alpha_e / price_i^c) \\ &= p^I (p_i^I R_i t + (1 - p_i^I) r t \alpha_e (1 + Q_0 - \bar{Q}_i(t)) / \chi_0). \end{aligned} \quad (13c)$$

At  $t = T_0$ , we have

$$\begin{aligned} \bar{Q}_i(T_0) &= p^I (p_i^I R_i T_0 + (1 - p_i^I) r T_0 \alpha_e / price_i^c) \\ &= p^I (p_i^I R_i T_0 + (1 - p_i^I) r T_0 \alpha_e (1 + Q_0 - \bar{Q}_i(T_0)) / \chi_0). \end{aligned} \quad (13d)$$

Alternatively to the previous definitions, a dynamic model for the consumed traffic volume can be defined as

$$\begin{aligned} Q_i(t+1) &= p^I (p_i^I R_i(t) \Delta t + (1 - p_i^I) e_i r \Delta t) \\ &= p^I (p_i^I R_i(t) \Delta t + (1 - p_i^I) r \Delta t \alpha_e \\ &\quad \times (1 + Q_0 - \bar{Q}_i(t)) / \chi_0) \end{aligned} \quad (15a)$$

where  $Q_i(t+1)$  is the volume of traffic transmitted in time  $(t, t+1)$ ,  $\Delta t$  is the duration of the transmission, and  $\bar{Q}_i(t)$  is the cumulative traffic transmitted until time  $t$  obtained by (15a) as

$$\bar{Q}_i(t) = \sum_{j=1}^t Q_i(j). \quad (15b)$$

Finally, the overall traffic transmitted during  $T_0$

$$\bar{Q}_i(T_0) = \sum_{t=1}^{T_0} Q_i(t). \quad (15c)$$

As performance measure, the next parameters are defined:

- The Internet *contract utilization*  $\xi_i$  defined as

$$\xi_i = \frac{\bar{Q}_i(T_0)}{Q_0} \quad (16)$$

where  $\bar{Q}_i(T_0)$  is the volume of traffic transmitted during the length of the contract,  $T_0$ , and  $Q_0$  is the initial amount of traffic volume available.

- The *contract price recovery*  $\varepsilon_i$  obtained as

$$\varepsilon_i = p^I \frac{1}{\chi_0} \left( \frac{\chi_0 p_i^I R_i T_0 + price_i^c (1 - p_i^I) e_i r T_0}{Q_0} \right) \quad (17)$$

where the first term is the percentage of price spent by terminal  $i$  on transmitting its own traffic and the second term is the percentage of price gained when acting as an AP.

The previous equations can be easily extended for other types of contracts (e.g.,  $T/W(r)/I(R)$ ,  $T/W(q)/I(R)$ , and  $T/W(q)/I(Q)$ ). This framework provides many opportunities for further extension of the models, especially in multioperator scenarios. The reward for serving as an access point may also depend on the capacity of the terminal, so that the weaker terminals (like smartphones) could be awarded more than PCs or conventional access points with higher capacity.

#### D. Security Investment

In this section, we are interested to evaluate how the user's security investment affects the performance and, consequently, the network utilization.

We assume that the user invests in security measures, such as purchasing software and configuring it on its system. This provides a security level  $L_i$  and brings an increased direct cost  $S_i$ . Higher cost results into higher security level. The security investment reduces the probability that the user would be vulnerable to an attack and also the probability that other users will be attacked. When the terminal acts as an AP for other users in its vicinity, it will benefit from its security investment, and the



probability of being elected as an AP,  $e_i$ , will be higher. The security level demanded by an adjacent user  $k$  when connecting to terminal  $i$  is

$$D_{ki} \geq L_k \quad (18)$$

so that its investment will not be wasted. Otherwise, the user will not be interested on that connection. Then, the probability that terminal  $i$  will be elected as an AP can be approximated as

$$e_i = \frac{\alpha_e}{price_i^c} p(L_i \geq D_{ki}) = \frac{\alpha_e}{price_i^c} p(L_i \geq L_k) \quad (19)$$

where  $e_i$  is inversely proportional to the  $price_i^c$  charged for the connection by terminal  $i$  and given by (14), and proportional to the probability that  $i$  will satisfy the security level demanded by user  $k$ .

If the difference between the security level of user  $k$  and terminal  $i$  is very large,  $L_i - L_k \gg 0$ , then  $i$  will charge an extra price,  $price_i^s = \alpha_s (S_i - S_k)$ , to serve user  $k$ , where  $\alpha_s$  is a proportionality constant. This cost will compensate that  $i$  will need to scan the system or take extra security measures after serving user  $k$ .

As result, the price that terminal  $i$  will charge to  $k$  for a *secure* connection is

$$price_i^{cs} = price_i^c + price_i^s. \quad (20)$$

The economic model discussed in Section IV-C should be modified to include  $e_i$  and the price defined by (19) and (20), respectively.

## V. DYNAMIC TRACKING OF THE OPTIMUM TOPOLOGY AND ARCHITECTURE BY GENETIC ALGORITHM

In a real network, when the traffic changes in time and space, an efficient mechanism is needed to reconfigure the optimum topology to the traffic variations in the network. The optimization problems defined in this section can be solved in a dynamic environment, where the topology in the observation instant  $t$  is denoted by  $\mathbf{T}^t$ . As already mentioned, the time to reconfigure the topology is limited by the network dynamics. For these reasons, a genetic algorithm is developed to track the changes in the optimum topology due to the traffic variations.

### A. GA

Genetic algorithm is a computational mechanism inspired by natural evolution where stronger individuals are more likely to survive in a competitive environment. GA has been shown to be a useful alternative to traditional search and optimization methods, especially for problems where the space of all potential solutions is too high to be searched exhaustively in any reasonable amount of time [6], [7].

The first step in GA is to encode the problem as a chromosome or a set of chromosomes that consist of several genes. Next, a pool of feasible solutions to the problem, called initial population, is created. A fitness value, calculated using a fitness function, is associated with each chromosome and indicates how good the chromosome is. Genetic operators' *selection*, *crossover*, and *mutation* operate on the population to generate a new generation of population, i.e., a new set of feasible solutions from the old ones. Good feasible solutions are selected with higher probability to the next generation, in line with the idea of survival of the fittest. As the algorithm continues and newer generations evolve, the quality of solutions improves. The success of GA and its applications are outlined in [38].

In this paper, we formulate a genome as a feasible topology  $\mathbf{T} \in \mathcal{T}$  that consists of a block of chromosomes  $\delta\mathbf{T}$  (partial topologies) that provide connection for  $N$  users to  $K$  access points in a scheduling period  $\Delta$ . We model the traffic variations in our DNA network by considering that the topology variations are due to the changes in the APs and users availability.

### B. Encoding and Initial Population

We encode the topologies  $\mathbf{T}$  as a block of chromosomes where each chromosome defines a partial topology  $\mathbf{T} = [\delta\mathbf{T}]$ ,  $\delta = 1, 2, \dots, \Delta$ , and  $\Delta$  is the scheduling length. Each entry of the chromosome  $\mathbf{T} = [\delta T_{ij}]$  represents a gene with genotype  $i$  (user) and phenotype  $j$  (access point). Hence, each gene defines a connection between user  $i$  and AP  $j$ .

We assume that the initial topology of the network is known and denoted by  $\mathbf{T}_0$ . Later on, we will provide details on the robustness of our algorithm to the initial state of the network. This topology needs to be modified accordingly based on the following possible changes in the network.

- 1) If a new user arrives to the network, then the new user will be initially assigned to its closest AP.
- 2) If a user leaves the network, then its entry will be zero in the topology matrix.
- 3) If a new AP appears in the network, then its closest user will be reassigned to this new AP
- 4) If a new AP leaves the network, then its users will be reassigned to the remaining closest AP.

There is also the possibility to detect more than one change at a time. This is the case when: 5) A user turns into an AP. In this case, the topology should be modified as indicated in 2) and 3); 6) an AP turns into a user. Then, the topology should be modified as indicated in 1) and 4).

The update of the initial topology  $\mathbf{T}_0$  results into  $\mathbf{T}_0^+$ . At any time instant  $t$ ,  $g$  topologies (genomes) are contained in the population pool  $P(t)$ . The initial population  $P(0)$  consists of topology  $\mathbf{T}_0^+$  and  $g-1$  topologies randomly generated,  $P(0) = \{\mathbf{T}_0^+, \mathbf{T}_1, \dots, \mathbf{T}_{g-1}\}$ . The next generation of population is obtained by applying *genetic operators* over the topologies of the current generation. The fitness function used to evaluate the topologies is given by the utility function defined by (6), (8), or (10).

### C. Genetic Operators

We consider the operations of *selection*, *crossover*, and *mutation*.

The *selection* operation consists of choosing 40% of the topologies that produce the highest fitness among the current population to survive to the next generation. Then, the rest of the new population is obtained by applying crossover and mutation operations to those selected topologies (30% topologies generated by each operator). This provides a compromise between the level of elitism and diversity to generate new topologies with the past of the generations.

The *crossover* operation consists of shifting two genes between two different chromosomes from the same genome to generate an offspring topology. In particular, we randomly choose one of the selected topologies at a time, and then we randomly choose two genes and apply the crossover to generate a new topology as

$$\delta_1 T(i_1, j_1) \rightleftharpoons \delta_2 T(i_2, j_2).$$



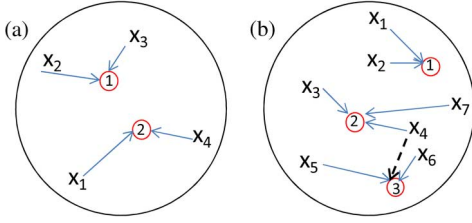


Fig. 3. Illustration of the genetic operators: (a) crossover and (b) mutation.

A topology can be repeatedly selected, but if the offspring topology results into an existing one, then it will be removed from the pool, and another topology will be randomly chosen. The new topology resulting from this operation will be always a feasible topology.

The purpose of this operation is to reduce the interference resulting from concurrent transmissions. In this case, we can move one of the interference users (the whole gene) to transmit in a different slot. This operation is illustrated with the scenario shown in Fig. 3(a), where initially users 1 and 2 share the first slot and users 3 and 4 transmit in the next slot. After the crossover, the genes 1 and 4 will be shifted, resulting into the following new topology:

$$\mathbf{T} = \begin{matrix} i \backslash AP & 1 & 2 & & 1 & 2 \\ & \delta=1 & & \delta=2 & & \\ 1 & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & & \\ 2 & & & & & \\ 3 & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & & \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & & \\ 4 & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} & & \end{matrix}; \mathbf{T}_{new} = \begin{matrix} i \backslash AP & 1 & 2 & & 1 & 2 \\ & \delta=1 & & \delta=2 & & \\ 1 & \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} & & \\ 2 & & & & & \\ 3 & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & & \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} & & \\ 4 & \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & & \end{matrix}$$

$$\delta T(1,2) \Leftrightarrow \delta T(4,2);$$

where users 2 and 4 will share the first slot and users 1 and 3 will transmit in the second slot. As we can see in Fig. 3(a), the interference in this topology will be reduced.

The *mutation* operation is performed to facilitate jumping of solutions to new unexplored regions of the search space. It consists of allocating the user to a more convenient AP. This can be achieved by mutating the phenotype of an individual gene. An example of the mutation operation is shown in Fig. 3(b), where the phenotype of user 4 is mutated from 2 to 3. The new topology results into the following:

$$\mathbf{T} = \begin{matrix} i \backslash AP & 1 & 2 & 3 & & 1 & 2 & 3 & & 1 & 2 & 3 \\ & \delta=1 & & \delta=2 & & \delta=3 & & & & \delta=1 & & \delta=2 & & \delta=3 \\ 1 & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} & & & & & & & & \\ 2 & & & & & & & & & & & & & & \\ 3 & & & & & & & & & & & & & & \\ 4 & \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & & & & & & & \\ 5 & & & & & & & & & & & & & & \\ 6 & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & & & & & & & \end{matrix}; \mathbf{T}_{new} = \begin{matrix} i \backslash AP & 1 & 2 & 3 & & 1 & 2 & 3 & & 1 & 2 & 3 \\ & \delta=1 & & \delta=2 & & \delta=3 & & & & \delta=1 & & \delta=2 & & \delta=3 \\ 1 & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} & & & & & & & & \\ 2 & & & & & & & & & & & & & & \\ 3 & & & & & & & & & & & & & & \\ 4 & \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & & & & & & & \\ 5 & & & & & & & & & & & & & & \\ 6 & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & & & & & & & & \end{matrix}$$

$$\delta T(4,2) \rightarrow \delta T(4,3);$$

#### D. Convergence and Complexity of GA

After a new generation of population is created by using the genetic operators described above, the fitness function is used

again to evaluate the generated topologies, and the process is repeated in the same fashion. As 40% of the best topologies are kept in the pool, the fitness of the best topology in each generation will always be better or equal than in the previous generation.

If we denote by  $f'_n(\mathbf{T}'_n)$  the best fitness function in generation  $n$  obtained for topology  $\mathbf{T}'_n$ , then GA converges to the solution when  $|f'_{n+1}(\mathbf{T}'_{n+1}) - f'_n(\mathbf{T}'_n)| \leq \varepsilon$ , where  $\varepsilon \simeq 0$ . The topology  $\mathbf{T}'_{n+1}$  will be the optimum topology  $\mathbf{T}^*$  with certain probability that we denote probability of success or success ratio,  $p_{sus}$ . This probability is obtained as the ratio between the number of times the optimum topology has been found with respect to the number of runs of the GA. In this process, exhaustive search is used to confirm the optimum topology. The optimum fitness function  $f^*$  is obtained for topology  $\mathbf{T}^*$  when  $p_{sus} = 1$ . If  $p_{sus} < 1$ , then  $|f^* - f'| > 0$  and the topology  $\mathbf{T}'_{n+1}$  is a suboptimum solution. To achieve the optimum solution, the size of the population  $g$  and the number of generations  $N_g$  considered for the GA must be adjusted to the size of the DNA. This will be shown in Section VI through simulations.

The complexity of the GA is given by the following parameters:

- total number of generated topologies  $G$  needed to obtain the optimum solution,  $G = g \cdot N_g$  where  $g$  is the number of topologies per generation and  $N_g$  is the number of generations;
- computational time that indicates the time it takes to obtain the optimum topology (CPU time).

#### E. 2L-AAC Scheme

The conventional admission control mechanisms aim at maintaining the required QoS by the users by limiting the number of new users that access the network at a given time. In this paper, we present a 2L-AAC protocol that regulates the access of new users and APs to the network in order to keep the level of utility per user  $U_i$  above certain threshold  $U_0$ .

If we assume that the network efficiency is measured by the utility defined by (6) and there is a user with utility  $U_i < U_0$ , then the 2L-AAC can improve  $U_i$  by performing one of the following actions.

- Allow a new AP  $k$ , located in a closer distance to user  $i$  than the actual AP  $j$ ,  $d_{ik} < d_{ij}$ , to access the network. In this case, the power consumption  $P_i$  will be reduced as the new AP is closer.
- Reduce the number of users that are transmitting to AP  $j$  so that the scheduling cycle will be reduced.
- Reduce the number of users that are sharing the slot with user  $i$ , which will increase the capacity  $c_{ij}(\mathbf{T})$ .
- All the previous options will also increase the overall utility. After there is a traffic change, the new topology should be reconfigured to provide the optimum performance according to (6).

If the utility also includes the cost as in (8) and the maximum utility at certain state of the network  $U_i < U_0$ , then the same actions as before can be applied. The only difference is that introducing a new AP will increase the cost, and this increment should compensate the gain on the utility. Otherwise, the new AP  $k$  may replace the actual AP  $j$ .

The illustration of the 2L-AAC scheme is shown in Fig. 4 where it is assumed that the arrival rate of users/APs,  $\lambda_m$ ,

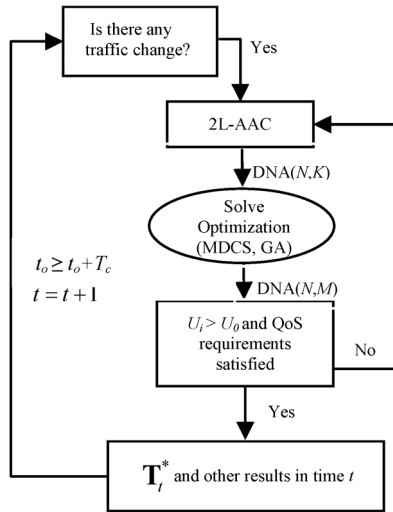


Fig. 4. Interaction of the 2L-AAC scheme and the optimization problem.

and the computational time,  $T_c$ , satisfy (5). The observation instant  $t_o$  is assumed to be a multiple of  $T_c$ .

### F. Implementation

The optimization problems will be run at the BS or equivalent network controller who will keep track on the existing traffic in each DNA cluster. The BS will assign the users to the most appropriate cluster, and the 2L-AAC mechanism will provide the access on a cluster basis.

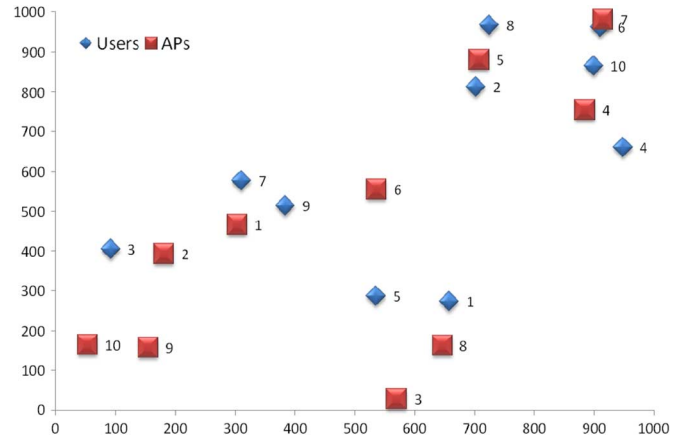
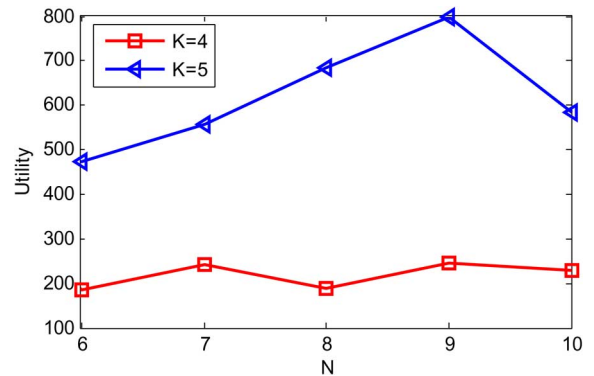
As result of the optimization problems, the optimum topology that provides the data for *intracluster* reallocation (handover) is obtained. The terminal status (user or AP) is communicated on the conventional uplink signaling (control) channel. Then, the network controller will assign each user to the most convenient AP according to a given utility function. We have assumed that the user-AP allocation is fixed during the scheduling cycle. Furthermore, we assumed that the computational time  $T_c$  needed to obtain the new optimum topology after a traffic change is  $T_c < 1/\lambda_m$  where  $\lambda_m$  is the call arrival rate. In this way, the new topology can track the network dynamics. The observation instant  $t_o$  is assumed to be  $t_o \geq T_c$  since the system cannot react to the changes faster than  $T_c$ . The size of the network cluster considered should be scaled accordingly to keep the computational complexity under that threshold.

At the DNA macro network, the resulting utility is obtained as  $U = \Gamma \cdot \sum_{i=1}^{\Gamma} U_i$ , where  $\Gamma$  is the reuse factor.

The *intercluster* handover may be handled by applying clustering/reclustering algorithms [39], [40] after a change in the traffic occurs.

## VI. PERFORMANCE EVALUATION

In this section, the performance of the network is evaluated through extensive computer simulations conducted using MATLAB. The scenario considered is shown in Fig. 5, where we assume that  $N$  nodes and  $K$  access points are randomly placed in an area of  $1000 \times 1000 \text{ m}^2$ . This scenario corresponds to a  $\text{DNA}(N, K)$ . The results for the DNA macro network

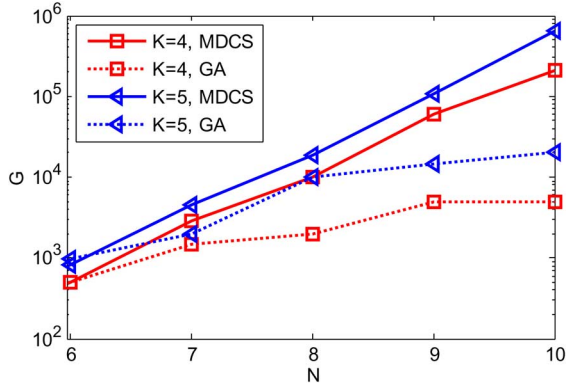
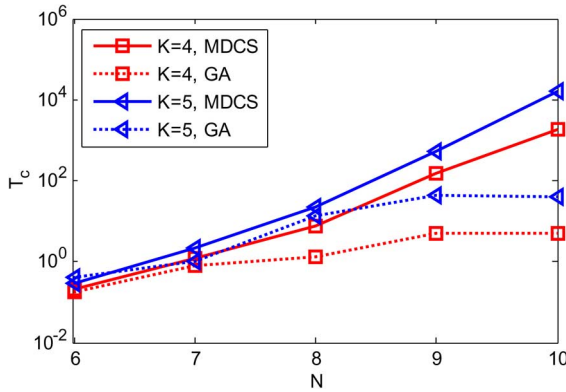
Fig. 5. Simulation scenario with index of user  $i = 1, \dots, N$  and AP  $j = 1, \dots, K$ .Fig. 6. Utility defined by (7) for the optimum topology versus  $N$ .TABLE I  
SIMULATION PARAMETERS

Simulation area	1000 x 1000 m <sup>2</sup>
$\alpha$	3
$N_0$	1
$\tau$	10mW
$p^j$	0.5
$\chi_0/Q_0$	1
$R_i/R_{ei}$	1, ..., 4
$\lambda_m$	0.01 calls/s

can be obtained as explained in Section V. The simulation parameters are summarized in Table I.

In Fig. 6, the utility obtained for the optimum topology is presented as a result of the optimization problem defined in (7) when there are  $K = 4$  and 5 APs, and  $N = 6, \dots, 10$  users. The location of those users and APs is shown in Fig. 5. When  $K = 4$  is used, APs with higher index from the figure are inactive. The same principle applies for  $K = 5$ . The utility is higher for  $K = 5$  as the transmission power and the number of slots needed to complete the transmission is lower than for  $K = 4$ . The optimization has been solved by MDCS and GA. To obtain the optimum topology by GA, the size of the population and the number of generations was scaled with the size of the DNA. The values used for the previous parameters will be explained later in Fig. 12.

In Figs. 7 and 8, the number of generated topologies  $G$  and the running time to obtain the optimum topology are shown,


 Fig. 7. Number of topologies  $G$  generated to solve (7) versus  $N$ .

 Fig. 8. Running time  $T_c$  needed to obtain the optimum topology by (7) versus  $N$ .

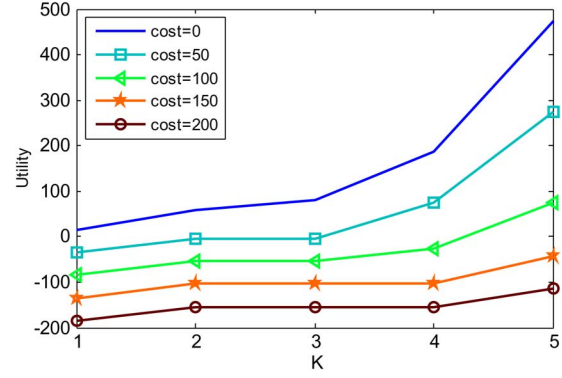
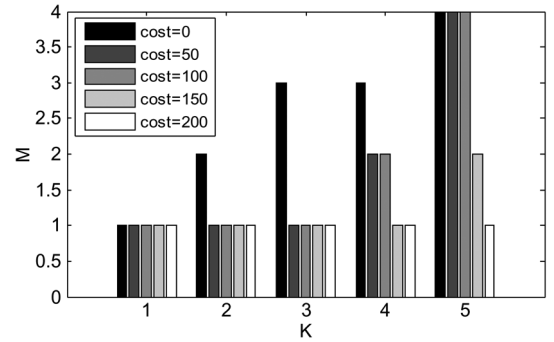
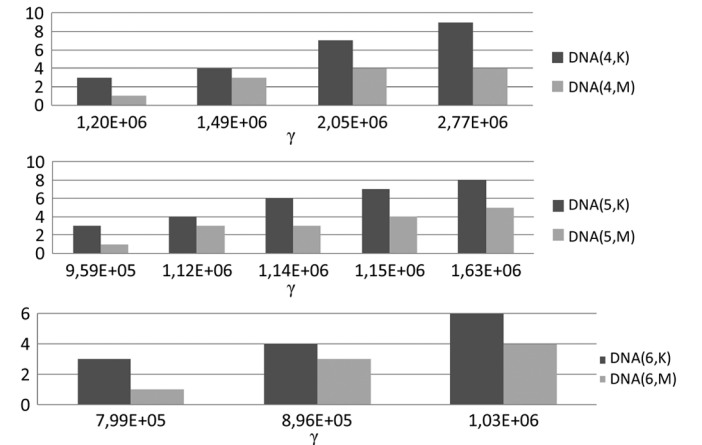
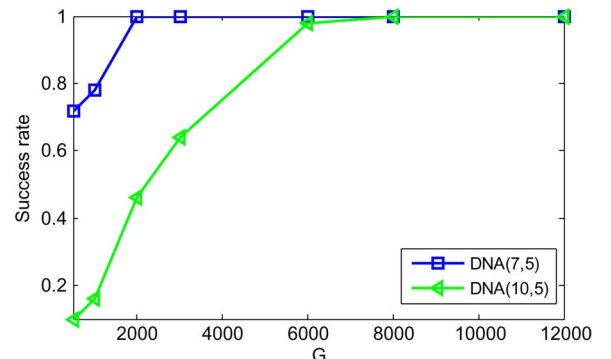
respectively, for MDCS and GA algorithm. A processor Intel Core i5-2400 CPU at 3.10 GHz with 8 GB RAM memory has been used for the simulations.

As we can see in those figures, for a given value of  $K$ , the improvement obtained by GA increases with  $N$ . It is worth noticing that the number of topologies generated by MDCS,  $G_{\text{MDCS}}$ , and thus the time,  $T_{c\text{MDCS}}$ , linearly increases with  $N$ . By GA, the increase of  $G_{\text{GA}}$  and  $T_{c\text{GA}}$  is more moderate. In particular, for DNA(4,8),  $G_{\text{GA}}$  is one order of magnitude smaller than  $G_{\text{MDCS}}$ . For each new user ( $N = 9$ ,  $N = 10$ ), one more order of magnitude is obtained as improvement by GA. The running time  $T_{c\text{GA}}$  for the scenarios considered is below 100 s, so for a typical value of the arrival rate  $\lambda_m = 0.01$  calls/s [41] and the size of the DNA considered, GA can track the changes in the optimum topology.

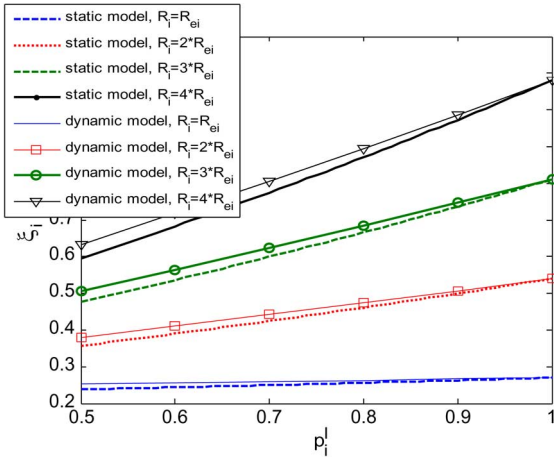
In Fig. 9, the utility is shown as result of the optimization problem defined by (8) when the cost = 0, ..., 200. The DNA considered consists of  $N = 6$  users and  $K = 1, \dots, 5$  available APs. The scenario is shown in Fig. 5 where users with index higher than 6 are inactive, and the same applies for the APs. As expected, the utility is lower for higher values of the cost and is higher for higher  $K$ . In Fig. 10, the optimum value of  $M$  is obtained for different values of the cost and  $K$ .

The results for the optimization problem (10) are presented in Fig. 11 for  $N = 4, \dots, 6$  and different values of  $K$ . The optimum  $M$  is obtained for each scenario for different values of the QoS constraint  $\gamma$ . As before, the scenarios considered are shown in Fig. 5.

In Fig. 12, the success ratio of GA is shown for the scenarios DNA(7,5) and DNA(10,5) versus the number of


 Fig. 9. Utility defined by (8) for the optimum topology versus  $K$  and  $N = 6$ .

 Fig. 10. Optimum  $M$  versus  $K$  for the optimization problem defined by (8).

 Fig. 11.  $M$  and  $K$  versus the QoS constraint  $\gamma$ .

 Fig. 12. Success rate versus  $G$  when the initial population is randomly chosen.

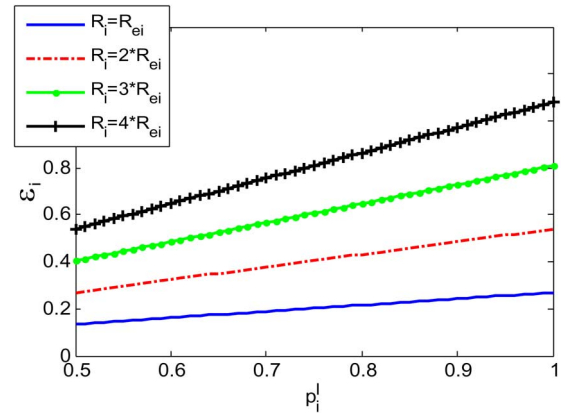
generated topologies  $G$ . In this case, we consider that the initial state of network is unknown, so this parameter is also

Fig. 13.  $\xi_i$  versus  $p_i^I$  for static and dynamic models of traffic volume.TABLE II  
GA PARAMETERS

$g$	$N_g$	$G$
10	50	500
20	50	1000
20	100	2000
30	100	3000
30	200	6000
40	200	8000
50	200	10000
60	200	12000

an indication of the robustness of GA to the initial state of the network. The initial population consists on a number of feasible topologies randomly generated. The number of topologies generated  $G$  is obtained as the product of the size of the population  $g$  and the number of generations  $N_g$ . In particular, the values of  $g$  and  $N_g$  used in the results are shown in Table II. The simulations have been generated 50 times, and the success ratio is defined as the number of times the optimum topology has been found with respect to the number of runs. We can see that for DNA(7,5),  $G = 2000$  topologies are generated where  $g = 20$  and  $N_g = 50$  are used to obtain the optimum topology with success ratio 1. In a bigger network, such as DNA(10,5),  $G = 8000$  are needed to obtain the optimum one with  $g = 40$  and  $N_g = 200$ .

In Fig. 13, the Internet contract utilization  $\xi_i$  defined by (16) is shown for different values of  $p_i^I$  and different ratios of  $R_i$  and  $R_{ei}$ . Both, static and dynamic models for the traffic volume are considered. The scenario used for this simulation is shown in Fig. 2 for the contract  $T/W(r)/I(Q)$ . The probability of Internet connection was set to  $p^I = 0.5$ , and the relation between the price  $\chi_0$  and the initial prepaid traffic volume  $Q_0$  has been normalized to 1, so that a monetary unit is charged for each unit of traffic volume. As result, we can see that the utilization  $\xi_i$  increases with  $p_i^I$  and with the rate of user  $i$ ,  $R_i$ . Moreover,  $\xi_i$  is larger when the dynamic model is used. The reason for this relies on the fact that the  $price_i^c$  used in the static model is the one obtained at the end of the contract  $T_0$ , so it is higher than for the dynamic model that is calculated in every time  $t$ . As a consequence, the probability that user  $i$  will be elected as AP,  $e_i$ , is lower, and thus there are more chances that the user will transmit its own traffic. For larger  $R_i$ , the higher is the difference between  $\xi_i$  for both models.

Fig. 14. Contract price recovery  $\varepsilon_i$  versus  $p_i^I$ .

The same scenario is used in Fig. 14 to present the contract price recovery  $\varepsilon_i$  versus  $p_i^I$ . As we can see in the figure,  $\varepsilon_i$  is larger for higher values of  $p_i^I$  and increases with the utilization of user  $i$ .

A set of simulations has been generated to show the effects of the security requirements on parameters  $\xi_i$  and  $\varepsilon_i$ . Those results are not included for space constraints. The conclusions obtained can be easily justified from (19) and (20). The higher the probability is that the user  $i$  will satisfy the security requirement of adjacent user  $k$ , the higher will be the probability of being elected as an AP. If the difference between the security level of user  $k$  and the one provided by the AP is very high, further security investment will be needed from the user  $k$  to be accepted by the AP.

Finally, some results are provided to show the efficiency of GA in a dynamic environment. In Table III, a number of scenarios are presented where the efficiency of GA in tracking the network dynamics is shown. The first column, DNA $t$ , represents the current network situation in time  $t$ . In the second column, DNA $t+1$  describes the network in the next time instant after the topology has changed. The description of the change is outline in the third column. The fourth column presents the value of the optimum utility (fitness) after the traffic changed. Columns 5 and 6 show the number of topologies generated by MDCS and GA, respectively, to obtain the optimum one. The computational time in seconds is shown in the last two columns. We can see that  $G_{GA} \ll G_{MDCS}$  for almost all scenarios. The only exception is when the size of DNA considered is small [e.g., DNA(4,3)]. In that case, as the number of possible combinations is low, the number of topologies generated by MDCS is lower than GA. For the rest of the scenarios, the order of improvement by GA regarding the number of topologies generated and the computational time can reach up to two orders of magnitude compared to MDCS. The number of topologies generated by MDCS,  $G_{MDCS}$ , increases exponentially with the size of DNA. The values of  $g$  and  $N_g$  to run the GA in this scenario resulted into a maximum  $G = 110$ . For this reason, the last three results of the table that correspond to DNA(7,4), DNA(8,4), and (DNA(9,4) have reached that number of topologies. The error in the optimum utility for these scenarios has been in the order of  $10^{-3}$ . It is worth mentioning that for DNA(9,4) the computational time  $T_c = 154.052 \text{ s} > 100 \text{ s}$ . Thus, the size of this DNA should be reduced in order to track the dynamics if the



TABLE III  
TOPOLOGY RECONFIGURATION SCENARIOS

DNA <sub>t</sub>	DNA <sub>t+1</sub>	Traffic changed	Optimum fitness	G <sub>MDCS</sub>	G <sub>GA</sub>	Computation time(sec) MDCS	Computation time(sec) GA
DNA(4,2)	DNA(4,3)	new AP	142.0225	14	30	0.0520	0.0450
DNA(4,3)	DNA(3,3)	exit user	279.5851	33	5	0.0800	0.0530
DNA(3,3)	DNA(4,3)	new user	142.0225	14	45	0.0520	0.0620
DNA(4,3)	DNA(4,4)	new AP	242.5647	43	15	0.0780	0.0510
DNA(4,4)	DNA(5,4)	new user	257.1448	153	15	0.1	0.0680
DNA(5,4)	DNA(6,4)	new user	186.6262	503	35	0.21	0.0690
DNA(6,4)	DNA(6,3)	exit AP	79.1647	189	65	0.171	0.0630
DNA(6,3)	DNA(6,4)	new AP	186.6262	503	85	0.21	0.0790
DNA(6,4)	DNA(7,4)	new user	241.4456	2919	110	1.2	0.0940
DNA(7,4)	DNA(8,4)	new user	188.144	9996	110	7.48	0.1030
DNA(8,4)	DNA(9,4)	new user	245.1063	58964	110	154.052	0.1090

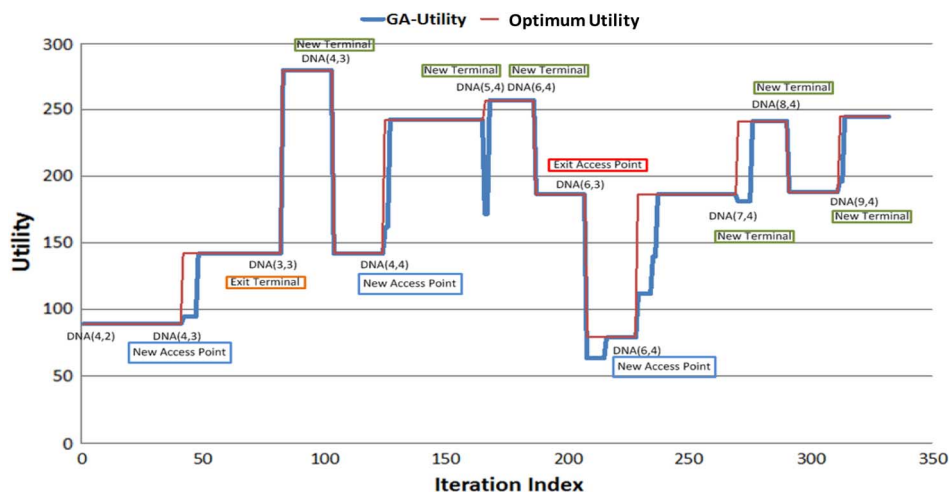


Fig. 15. Dynamic topology and architecture reconfiguration scenarios.

arrival rate is set to  $\lambda_m = 0.01$  calls/s. For the same scenarios considered in Figs. 7 and 8,  $G$  and  $Tc$  are now much lower. This is because, in these scenarios, the previous state of the network, given by topology  $T_0$ , is known (topology reconfiguration), while in Figs. 7 and 8, the optimization was run starting from any random topology.

The representation of the dynamics is shown in Fig. 15 for the scenarios described in Table III. In this chart, we can see how GA tracks the changes in the network and reaches the optimum value of the utility computed by MDCS.

## VII. CONCLUSION

In this paper, we have presented and analyzed a new network paradigm. In this concept, certain classes of terminals when connected to the Internet (wired/wireless) can serve as an AP for other users in its vicinity. As the location of users and APs changes in time, this results into a dynamic network architecture. A DNA optimization framework is presented to minimize the number of APs needed to satisfy the QoS requirements from the users and, thus, reduce the network cost. Then, an economic model to incentivize the users to act as APs is proposed to efficiently use the capacity provided by the network operator for other users' transmissions.

The economic model and security investment should be considered only as possible (although real and interesting) examples, and further work in this segment is expected in the future.

When the users' and APs' availability changes, a genetic algorithm is presented to dynamically reconfigure the optimum topology and adjust it to the traffic changes in the network, so that the best performance is guaranteed.

Simulation results have shown that the performance improvement of GA compared to exhaustive MDCS can reach up to two orders of magnitude in terms of number of topologies generated and running time to obtain the optimum topology. We have also shown the impact of the cost on the election of the AP and the number of APs needed to satisfy the QoS constraints.

This work presents a framework for a novel architecture where different aspects can be further elaborated. As possible extensions of this work, we are planning to consider terminals equipped with multiple antennas and develop beamforming schemes to select the most appropriate APs that satisfy the QoS requirements and maximizes throughput.

## REFERENCES

- [1] M. Uyar, J. Zheng, M. Fecko, S. Samtani, and P. Conrad, "Evaluation of architectures for reliable server pooling in wired and wireless environments," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 1, pp. 164–175, Jan. 2004.
- [2] A. Akella *et al.*, "Self-management in chaotic wireless deployments," in *Proc. 11th Annu. Int. Conf. Mobile Comput. Netw.*, 2005, pp. 185–199.
- [3] N. Funabiki *et al.*, "A proposal of an active access-point selection algorithm in wireless mesh networks," in *Proc. NBiS*, 2011, pp. 112–117.
- [4] T. Nakamura *et al.*, "Trends in small cell enhancements in LTE advanced," *IEEE Commun. Mag.*, vol. 51, no. 2, pp. 98–105, Feb. 2013.

- [5] Spark, "Setup device as wireless modem for PC—Nokia Lumia 920," [Online]. Available: [http://help.telecom.co.nz/app/answers/detail/a\\_id/30335/~/-setup-device-as-wireless-modem-for-pc-nokia-lumia-920](http://help.telecom.co.nz/app/answers/detail/a_id/30335/~/-setup-device-as-wireless-modem-for-pc-nokia-lumia-920)
- [6] L. Davis, *Handbook of Genetic Algorithms*. New York, NY, USA: Van Nostrand Reinhold, 1991.
- [7] B. Lorenzo and S. Glisic, "Optimal routing and traffic scheduling for multihop cellular networks using genetic algorithm," *IEEE Trans. Mobile Comput.*, vol. 12, no. 11, pp. 2274–2288, Nov. 2013.
- [8] W.-Z. Guo *et al.*, "Particle swarm optimization for the degree-constrained MST problem in WSN topology control," in *Proc. Int. Conf. Machine Learning Cybernetics*, Jul. 2009, vol. 3, pp. 1793–1798.
- [9] S. Yang *et al.*, "Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile ad hoc networks," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 40, no. 1, pp. 52–63, Jan. 2010.
- [10] E. Bonabeau, M. Dorigo, and T. Swarm, *Intelligence: From Natural to Artificial Systems*. Oxford, U.K.: Oxford Univ. Press, 1999.
- [11] R. Cohen and G. Nakibly, "A traffic engineering approach for placement and selection of network services," *IEEE/ACM Trans. Netw.*, vol. 17, no. 2, pp. 487–500, Apr. 2009.
- [12] Y. Fukuda, T. Abe, and Y. Oie, "Decentralized access point selection architecture for wireless LANs," in *Proc. 3rd Wireless Telecommun. Symp.*, 2004, pp. 137–145.
- [13] B. Kauffmann *et al.*, "Measurement-based self organization of interfering 802.11 wireless access networks," in *Proc. IEEE INFOCOM*, 2007, pp. 1451–1459.
- [14] S. Vasudevan, K. Papagiannaki, C. Diot, J. Kurose, and D. Towsley, "Facilitating access point selection in IEEE 802.11 wireless networks," in *Proc. 5th ACM SIGCOMM Conf. Internet Meas.*, 2005, pp. 293–298.
- [15] A. Mishra, V. Brik, S. Banerjee, A. Srinivasan, and W. Arbaugh, "A client-driven approach for channel management in wireless LANs," in *Proc. IEEE INFOCOM*, 2006, pp. 1–12.
- [16] J. Geier, "Assigning 802.11b access point channels," Wi-Fi Planet, Feb. 11, 2002 [Online]. Available: <http://www.wi-fiplanet.com/tutorials/article.php/972261>
- [17] Y. Zhu *et al.*, "On optimal physical carrier sensing: Theoretical analysis and protocol design," in *Proc. IEEE INFOCOM*, 2007, pp. 2351–2355.
- [18] Y. Hua, Y. Zhu, and Z. Niu, "A channel assignment scheme in high density WLANs to mitigate pseudo capture effect," in *Proc. IEEE GLOBECOM*, 2007, pp. 4796–4800.
- [19] Y. Bejerano, S.-J. Han, and L. Li, "Fairness and load balancing in wireless LANs using association control," *IEEE/ACM Trans. Netw.*, vol. 15, no. 3, pp. 560–573, Jun. 2007.
- [20] N. Blefari-Melazzi, D. D. Sorte, M. Femminella, and G. Reali, "Autonomic control and personalization of a wireless access network," *Comput. Netw.*, vol. 51, no. 10, pp. 2645–2676, 2007.
- [21] H. Nishiyama, H. Yamada, H. Yoshino, and N. Kato, "A cooperative user-system approach for optimizing performance in content distribution/delivery networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 2, pp. 476–483, Feb. 2012.
- [22] M. Hong, A. Garcia, J. Barrera, and S. G. Wilson, "Joint access point selection and power allocation for uplink wireless networks," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3334–3347, Jul. 2013.
- [23] I. Malanchini, M. Cesana, and N. Gatti, "Network selection and resource allocation games for wireless access networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 12, pp. 2427–2440, Dec. 2012.
- [24] R. Trestian *et al.*, "Game theory-based network selection: Solutions and challenges," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 1212–1231, 4th Quart., 2012.
- [25] K. Mittal, E. M. Belding, and S. Suri, "A game-theoretic analysis of wireless access point selection by mobile users," *Comput. Netw.*, vol. 31, no. 10, pp. 2049–2062, 2008.
- [26] L.-H. Yen, J.-J. Li, and C.-M. Lin, "Stability and fairness of AP selection games in IEEE 802.11 access networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 1150–1160, Mar. 2011.
- [27] F. Xu, C. Tan, Q. Li, G. Yan, and J. Wu, "Designing a practical access point association protocol," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.
- [28] J. Musacchio and J. Walrand, "WiFi access point pricing as a dynamic game," *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 289–301, Apr. 2006.
- [29] M. Q. Khan and S. H. Andresen, "PoA selection in 802.11 networks using Media Independent Information Server (MIIS)," in *Proc. WAINA*, 2012, pp. 454–459.
- [30] D. Niyato and E. Hossain, "Wireless broadband access: WiMAX and beyond—Integration of WiMAX and WiFi: Optimal pricing for bandwidth sharing," *IEEE Commun. Mag.*, vol. 45, no. 5, pp. 140–146, May 2007.
- [31] V. A. Siris, "Cell dimensioning in the CDMA uplink based on economic modelling," *Eur. Trans. Telecommun.*, vol. 18, no. 4, pp. 427–433, Jun. 2007.
- [32] M. H. Manshaei *et al.*, "On wireless social community networks," in *Proc. IEEE INFOCOM*, 2008, pp. 1552–1560.
- [33] N. Weragama, J. Jun, J. Mitro, and D. P. Agrawal, "Modeling and performance of a mesh network with dynamically appearing and disappearing femtocells as additional internet gateways," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 5, pp. 1278–1288, May 2014.
- [34] S.-Y. Yun, Y. Yi, D.-H. Cho, and J. Mo, "The economic effects of sharing femtocells," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 596–606, Apr. 2012.
- [35] L. Duan, J. Huang, and B. Shou, "Economics of femtocell service provision," *IEEE Trans. Mobile Comput.*, vol. 12, no. 11, pp. 2261–2273, Nov. 2013.
- [36] B. Lorenzo and S. Glisic, "Context aware nano scale modeling of multicast multihop cellular network," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 359–372, Apr. 2013.
- [37] E. Arıkan, "Some complexity results about packet radio networks," *IEEE Trans. Intell. Transport. Syst.*, vol. IT-30, no. 4, pp. 910–918, Jul. 1984.
- [38] J. J. Grefenstette, "Genetic algorithms for changing environments," in *Proc. 2nd Int. Conf. Parallel Problem Solving Nature*, 1992, pp. 137–144.
- [39] Z. Khan, S. Glisic, L. A. DaSilva, and J. Lehtomäki, "Modeling the dynamics of coalition formation games for cooperative spectrum sharing in an interference channel," *IEEE Trans. Comput. Intell. AI in Games*, vol. 3, no. 1, pp. 17–31, Mar. 2010.
- [40] E. Karami and S. Glisic, "Stochastic model of coalition games for spectrum sharing in large scale interference channels," in *Proc. IEEE ICC*, Jun. 2011, pp. 1–6.
- [41] Y. H. Tam, R. Benkoczi, H. S. Hassanein, and S. G. Akl, "Channel assignment for multihop cellular networks: Minimum delay," *IEEE Trans. Mobile Comput.*, vol. 9, no. 7, pp. 1022–1034, Jul. 2010.

**Alireza Shams Shafiq**, (S'13) photograph and biography not available at the time of publication.

**Beatriz Lorenzo**, (M'08) photograph and biography not available at the time of publication.

**Savo Glisic**, (M'90–SM'94) photograph and biography not available at the time of publication.

**Jordi Pérez-Romero**, (S'98–A'01–M'04) photograph and biography not available at the time of publication.

**Luiz A. DaSilva**, (S'97–M'98–SM'00) photograph and biography not available at the time of publication.

**Allen B. MacKenzie**, (S'96–M'03–SM'08) photograph and biography not available at the time of publication.

**Juha Rönning**, photograph and biography not available at the time of publication.