A Network-assisted Approach for RAT Selection in Heterogeneous Cellular Networks

Melhem El Helou, *Member, IEEE,* Marc Ibrahim, Samer Lahoud, Kinda Khawam, Dany Mezher and Bernard Cousin, *Member, IEEE*

***Abstract*—When several radio access technologies (*e.g.,* HSPA, LTE, WiFi and WiMAX) cover the same region, deciding to which one mobiles connect is known as the Radio Access Technology (RAT) selection problem. To reduce network signaling and processing load, decisions are generally delegated to mobile users. Mobile users aim to selfishly maximize their utility. However, as they do not cooperate, their decisions may lead to performance inefficiency. In this paper, to overcome this limitation, we propose a network-assisted approach. The network provides information for the mobiles to make more accurate decisions. By appropriately tuning network information, user decisions are globally expected to meet operator objectives, avoiding undesirable network states. Deriving network information is formulated as a Semi-Markov Decision Process (SMDP), and optimal policies are computed using the *Policy Iteration* algorithm. Also, and since network parameters may not be easily obtained, a reinforcement learning approach is introduced to derive what to signal to mobiles. The performances of optimal, learning-based, and heuristic policies, such as blocking probability and average throughput, are analyzed. When tuning thresholds are pertinently set, our heuristic achieves performance very close to the optimal solution. Moreover, although it provides lower performance, our learning-based algorithm has the crucial advantage of requiring no prior parameterization.**

***Index Terms*—Radio access technology selection, semi-Markov decision process, reinforcement learning, heterogeneous cellular networks.**

I. Introduction

THE demand for high-quality and high-capacity radio networks is continuously increasing. It has been reported that global mobile data traffic grew by 81 percent in 2013 [1]. Furthermore, monthly mobile traffic is forecast to surpass 15 exabytes by 2018, nearly 10 times more than in 2013 [1]. Along with this impressive growth, mobile operators are urged to intelligently invest in network infrastructure. They also need to reconsider their flat-rate pricing models [2], seeking positive return-on-investment.

Melhem El Helou, Marc Ibrahim and Dany Mezher are with the Ecole Supérieure d'Ingénieurs de Beyrouth (ESIB), Faculty of Engineering, Saint Joseph University of Beirut, P.O. BOX 1514, Riad El Solh, Beirut 1107 2050, Lebanon (e-mail: melhem.helou@usj.edu.lb; marc.ibrahim@usj.edu.lb; dany.mezher@usj.edu.lb).

Samer Lahoud and Bernard Cousin are with the Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA) laboratory, University of Rennes 1, 35042 Rennes, France (e-mail: samer.lahoud@irisa.fr; bernard.cousin@irisa.fr).

Kinda Khawam is with the PRiSM laboratory, University of Versailles, 45 Avenue des Etats-Unis, 78035 Versailles, France (e-mail : kinda.khawam@prism.uvsq.fr).

Manuscript received July 21, 2014; revised December 18, 2014.

To cope with this huge demand for capacity, next-generation networks rely on densely deployed base stations with hierarchical cell structures [3] (*i.e.,* macro, micro, pico and femto cells). A cost-effective solution is to use existing radio access technologies (RATs). Upcoming 5G networks are thus being devised with the vision of heterogeneity [3]. Various RATs, including 3GPP and IEEE families, are being integrated and jointly managed. Deciding to which one mobiles connect is known as the RAT selection problem.

When intelligence is pushed to the network edge, rational users select their RAT in a way to selfishly maximize their utility. However, as users have no information on network load conditions, their decisions may be in no one long-term interest, causing performance degradation. Moreover, delegating decisions to the network optimizes overall performance, but at the cost of increased network complexity, signaling and processing load.

A real challenge is to design a RAT selection method that jointly enhances network performance and user experience, while signaling and processing burden remains reduced. We propose, in this paper, a network-assisted approach. The network provides information for the mobiles to make appropriate RAT selections. More precisely, mobile users select their RAT depending on their individual needs and preferences, as well as on the cost and QoS parameters signaled by the network. By appropriately tuning network information, user decisions are globally expected to meet operator objectives, avoiding undesirable network states. Thus, our approach enables self-optimization, a key feature of self-organizing networks [4].

Our network-assisted approach has the particularity to involve two inter-dependent decision-making processes. The first one, on the network side, consists in deriving appropriate network information, so as to guide user decisions in a way to satisfy operator interests. The second one, where individual users combine their needs and preferences with the signaled network information, consists in selecting the RAT to be associated with, in a way to maximize user utility. Since, in their turn, user individual decisions influence the upcoming network information, the two decision makings are regarded to be inter-dependent. Thus, RAT selections dynamically involve operator objectives, and user needs and preferences.

To maximize long-term network performance, network information should depend not only on present load conditions, but also on expected future demands. In this paper, deriving network information is formulated as a Semi-Markov Decision Process (SMDP) [5]. The aim is to dynamically meet operator objectives, while mobiles maximize their own utility. Also,

when network parameters are not perfectly known, a reinforcement learning approach is introduced to derive what to signal to mobiles. The network learns user needs, preferences and decision-making algorithms through interacting with them.

II. Related Work

A key driver for heterogeneous cellular networks is to enable traffic class-aware optimal coverage, capacity, and reliability with low cost and energy consumption. To maximize user experience, mobiles select their serving RAT depending on user needs and preferences. As user experience does not only depend on their own decisions, but also on the decisions of other mobiles, game theory is used as a theoretical framework to model user interactions in [6]–[12]. Players (*i.e.,* the individual users) try to reach a mutually agreeable solution, or equivalently a set of strategies they unlikely want to change. However, the convergence time to the equilibrium assignment seems to be long [7]. In [13]–[21], multi-criteria decision-making methods, including Simple Additive Weighting (SAW), Multiplicative Exponent Weighting (MEW), Grey Relational Analysis (GRA) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) are presented. They capture the heterogeneity of decision parameters (*e.g.,* QoS, cost, energy and security parameters). Users with widely varying requirements gather their QoS information (*e.g.,* peak throughput when connected alone to a cell), calculate decision metrics, and select their RAT accordingly. In [16], [21], [22], fuzzy logic is also used to deal with the imprecise information of some criteria and user preferences.

Since decisions are delegated to mobile users, network operations remain reduced. Furthermore, decisions can easily involve user needs and preferences, and various mobile-terminal-related parameters. However, as mobiles do not cooperate, their decisions potentially lead to performance inefficiency. To overcome this limitation, we introduce in this paper a network-assisted approach. The network provides cost and QoS parameters to assist mobile users in their decisions. This decisional information reflects network load conditions and other potential operator concerns. For instance, by signaling appropriate network information, mobiles are globally pushed to some base stations, while others are switched to sleep mode to save energy. RAT selections are then expected to meet operator objectives, while mobiles individually maximize their utility.

The basic idea of our approach was first presented in [23], where heuristic policies are introduced to dynamically tune network information as a function of the load conditions. In the present contribution, deriving network information is formulated as a Semi-Markov Decision Process. We first define network states, actions, state dynamics and rewards. An optimal policy (*i.e.,* network information to signal in each state) is then derived through the *Policy Iteration* algorithm. As a matter of fact, transitions between network states depend not only on network actions, user arrival and departure rates, but also on user needs, preferences and decison-making algorithms. When all these parameters can not be easily obtained in constantly varying networks, an optimal policy is constructed through learning. Among the different existing reinforcement learning (RL) algorithms, we select the Q-learning method for its simplicity.

Furthermore, SMDP and Q-learning have been widely employed in RAT selection. In [24]–[30], RAT selection is modeled as a semi-Markov decision process. Network elements gather information about individual users, particularly their QoS needs and radio conditions. They decide on user serving cells in a way to maximize long-term network reward, without aligning with user preferences. However, network complexity, processing, and signaling load are drastically increased. Moreover, in [31]–[33], mobiles learn selection decisions through trial-and-error interaction with their dynamic environment. Yet, because of the non-cooperative behavior of mobile users, their performance may be degraded. In this paper, SMDP and Q-learning are used in a network-assisted approach. They enable the network to derive information for the mobiles to make decisions.

The rest of this paper is organized as follows: The network model is presented in section III. Section IV explains our hybrid decision framework. The SMDP and the *Policy Iteration* algorithm are introduced in section V. Section VI describes the Q-learning algorithm. Performance results are analyzed in section VII. Section VIII concludes the paper.

III. Network Model

A. Network Topology

Consider a heterogeneous cellular network composed of two OFDM(A)-based radio access technologies. Let x_1 and x_2 designate the two serving RATs within the network. Although our method adapts to different deployment scenarios, we focus on a realistic and cost effective one where the two RATs base stations are co-localized. The modulation and coding scheme, that can be assigned to a user connected to RAT x, differs depending on its radio conditions in the cell, more precisely on its signal-to-noise ratio denoted by SNR^x. As the number of possible modulation and coding schemes is limited, we decompose the cell into N_Z^x zones with homogeneous radio characteristics [28]–[30]. Users in zone Z_k^x, $k = 1, ..., N_Z^x$, are assumed to have a signal-to-noise ratio between δ_k^x and δ_{k-1}^x, and then to use $mod^x(k)$ with $cod^x(k)$ as modulation and coding scheme:

$$\begin{cases} \text{none} & \text{if } SNR^x(k) < \delta^x_{N_Z^x}, \\ (mod^x_{N_Z^x}, cod^x_{N_Z^x}) & \text{if } \delta^x_{N_Z^x} \leq SNR^x(k) < \delta^x_{N_Z^x-1}, \\ \cdots & \\ (mod^x_1, cod^x_1) & \text{if } \delta^x_1 \leq SNR^x(k) < \delta^x_0 = \infty. \end{cases} \quad (1)$$

where $\delta^x_{N_Z^x}$ is the minimum signal-to-noise ratio, that allows transmission at the lowest throughput, given a target error probability.

1) Cell Decomposition: Because of fading effects, radio conditions are time-varying. User signal-to-noise ratio can take all possible values, leading to different modulation and coding schemes. However, as RAT selections are made for a sufficiently long period of time (*e.g.,* session duration, user dwell time in the cell), users are distributed over logical zones

depending on their average radio conditions, rather than on their instantaneous ones.

Another approach is found in [28], where an analytical radio model, that accounts for interference, path loss, and Rayleigh fading, is used. It has been demonstrated that users need to be situated at $r_k \in [R^x_{k-1}, R^x_k[$ from their base stations, so as to have a signal-to-noise ratio between δ^x_k and δ^x_{k-1}, with at least a high probability $\mathbb{P}_{th}$. This means that the cell may be divided into concentric rings, as illustrated in Fig. 1, and mobiles in ring Z^x_k will use $mod^x(k)$ with $cod^x(k)$ as modulation and coding scheme, with at least a high probability $\mathbb{P}_{th}$. Further, to define the different rings, the distances R^x_k have been analytically derived, mainly as a function of δ^x_k, $\mathbb{P}_{th}$, and radio model parameters.

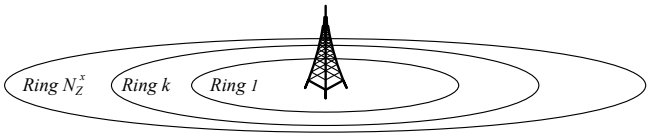

Fig. 1. RAT x cell divided into N^x_Z concentric rings

In what follows, and for the sake of simplicity, users in a same zone are assumed to have the same average signal-to-noise ratios.

B. Network Resources

The radio resource is divided into time-frequency resource units (RUs). Users in zone Z_k can transmit up to $b^x(k)$ bits per resource unit, when connected to RAT x:

$$b^x(k) = N^x_s \cdot N^x_f \cdot log_2[sz(mod^x(k))] \cdot R(cod^x(k)) \cdot (1 - BLER) \quad (2)$$

where N^x_s and N^x_f respectively denote the number of OFDM symbols and subcarriers per RU, $sz(mod^x(k))$ the constellation size of $mod^x(k)$, $R(cod^x(k))$ the coding rate of $cod^x(k)$ and $BLER$ the block error rate obtained as a function of the user signal-to-noise ratio.

In the time dimension, resources are organized into frames of length T^x. When RAT x allocates N_{res} resource units per frame to a user in zone Z_k, its average throughput d is given by:

$$d = \frac{N_{res} \cdot b^x(k)}{T^x} \quad (3)$$

C. Traffic Model

Users belong to N_C traffic classes. In our work, we focus on both streaming ($c = 1$) and elastic ($c = 2$) traffic classes. Class c arrivals, in zone Z_k, follow a Poisson process of rate $\Lambda(k, c)$. We assume that streaming sessions have an average long-term throughput of R_{av}. Yet, to improve their content quality, they can benefit from throughputs up to R_{max}. Their duration is considered to be exponentially distributed with a mean of $1/\mu_1$.

Moreover, elastic sessions adapt to resource availability. Their needs are expressed as comfort throughput denoted by R_c, and their size is assumed to be exponentially distributed with a mean of L bytes. However, in addition to their size, their service rate μ_2 also depends on their average throughputs.

IV. Hybrid Decision Framework

A. Network Information

Periodically or upon user request, network information is sent to all mobiles using the logical communication channel (*i.e.,* radio enabler) proposed by the IEEE 1900.4 standard [34]. This logical channel allows information exchange between the Network Reconfiguration Manager (NRM), on the network side, and the Terminal Reconfiguration Manager (TRM), on the mobile-terminal side (Fig. 2). The purpose is to improve resource utilization and user experience in heterogeneous cellular networks.

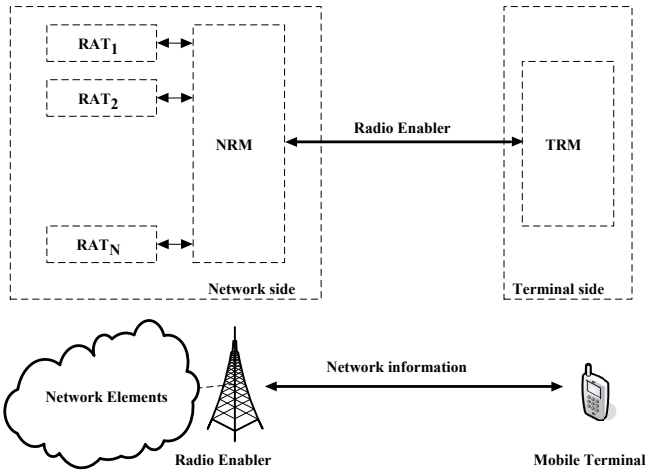

Fig. 2. Hybrid IEEE 1900.4 network architecture

In our work, by appropriately tuning network information, user decisions are globally expected to meet operator objectives, avoiding undesirable network states. Mobiles make final decisions regarding selection of their most appropriate RAT. As a matter of fact, based on their needs and preferences, as well as on the signaled network information, mobile users choose their RAT in a way to selfishly maximize their utility.

For RAT x, the network broadcasts QoS parameters, denoted by $d_{min}(x)$ and $d_{max}(x)$, and the cost to pay per amount of traffic, denoted by $cost(x)$ [23]. Mobiles are actually guaranteed an average minimum throughput, namely $d_{min}(x)$, and have priority to be allocated up to an average maximum throughput, namely $d_{max}(x)$. As $d_{min}(x)$ and $d_{max}(x)$ are derived for a generic user with the most robust modulation and coding scheme, individual users need to deduce their own QoS parameters. For that, mobiles in zone Z_k multiply the QoS parameters, signaled by the network, with their modulation and coding gain, denoted by $g(k)$.

B. RAT Selection

Using the satisfaction-based multi-criteria decision-making method we have introduced in [14], mobiles compute a utility function for each of the serving RATs, and select the one with

the highest score. This utility depends on user radio conditions, needs and preferences (*i.e.,* traffic class, throughput demand, QoS-maximizing or cost-minimizing preferences), as well as on the cost and QoS information sent by the network.

In our work, when $cost(x)$ is maintained fixed, $d_{min}(x)$ and $d_{max}(x)$ are dynamically tuned trying to globally control user decisions. Let N_I^x be the number of possible ($d_{min}(x)$, $d_{max}(x)$) couples, that may be signaled to incite mobile users to join RAT x. In the next section, selecting the ($d_{min}(x)$, $d_{max}(x)$) couple to be broadcasted, for each RAT x, is formulated as a Semi-Markov Decision Process (SMDP). The goal is to dynamically optimize the long-term discounted network reward, while mobiles maximize their own utility.

V. Semi-Markov Decision Process

At each user arrival or departure, signaled network information may have to vary. In this section, the SMDP is used to dynamically decide of the QoS parameters in a way that optimizes the long-term network reward. We first start by defining network states, actions, state dynamics and rewards. Next, using the *Policy Iteration* algorithm, we compute the optimal solution.

A. Network States

We define a state of RAT x to be the ($N_Z \times N_C \times N_I^x$)-tuple $n^x(t)$ for $\{k = 1, ..., N_Z, c = 1, ..., N_C,\ \ i = 1, ..., N_I^x\}$:

$$n^x(t) = (n^x(k, c, i, t)),$$

where $n^x(k, c, i, t)$ is a stochastic process representing the number of class c users in zone Z_k, that have the i^{th} ($d_{min}(x)$, $d_{max}(x)$) couple, at time t. In the remaining, we omit t as we assume stationarity.

To protect ongoing sessions, an admission control policy is applied: new arrivals may join RAT x, with the i^{th} ($d_{min}(x)$, $d_{max}(x)$) couple, to the extent that RAT x available resources are enough to meet their d_{min}, while not compromising the QoS guarantees of ongoing ones. Consequently, the set of admissible states in RAT x is:

$$\mathcal{N}_a^x = \Bigg\{ n^x \in \mathbb{N}^{N_Z \times N_C \times N_I^x} \mid \sum_{k=1}^{N_Z} \sum_{c=1}^{N_C} \sum_{i=1}^{N_I^x} n^x(k, c, i) \cdot N_{min}^x(i) \leq N_{total}^x \Bigg\} \quad (4)$$

where $N_{min}^x(i)$ is the number of RUs necessary to guarantee the d_{min} of the i^{th} QoS parameters couple, and N_{total}^x is the total number of RUs used for data transmission in RAT x.

Let the ($N_Z \times N_C \times N_I^{x_1}$ + $N_Z \times N_C \times N_I^{x_2}$)-tuple $s = (n^{x_1}, n^{x_2})$ be the state of the heterogeneous network, defined as the concatenation of RAT x_1 and RAT x_2 substates. The state space $\mathcal{S}$ of the network is then defined as:

$$\mathcal{S} = \{s = (n^{x_1}, n^{x_2}) \mid n^{x_1} \in \mathcal{N}_a^{x_1}, n^{x_1} \in \mathcal{N}_a^{x_2}\}$$

B. Network Actions

In each state, an action is taken by the network: QoS incentives to join serving RATs are derived. An action a is the quadruple defined by $a = (d_{min}(x), d_{max}(x))$, $x \in \{x_1, x_2\}$, where $d_{min}(x)$ and $d_{max}(x)$ represent the QoS parameters of RAT x, for the most robust modulation and coding scheme. Based on their needs (*e.g.,* traffic class, throughput demand) and preferences, as well as on their modulation and coding scheme (*i.e.,* geographical position), users act differently upon these actions. Refer to appendix A for more details.

Obviously, $N_I^{x_1} \cdot N_I^{x_2}$ actions are possible. However, given a state $s = (n^{x_1}, n^{x_2})$, not all actions are feasible. We then denote by $\mathcal{A}$ the set of all possible actions, and by $\mathcal{A}(s) \subset \mathcal{A}$ the subset of feasible actions in state s.

When both RATs provide no QoS incentives (*i.e.,* $d_{min}(x_1) = d_{max}(x_1) = d_{min}(x_2) = d_{max}(x_2) = 0$), action a is blocking and new arrivals are rejected.

C. State Dynamics

As the network does not completely control individual decisions, transitions between network states do not only depend on network actions, user arrival and departure rates, but also on user needs and preferences. The decision-making on the mobile side, using a multi-criteria decision-making method, then has a probabilistic impact on the transition rates.

Let $p^x(k, c, a)$ represent the probability that class c users in zone Z_k select RAT x, when action a is adopted. As action a may be blocking, $p^{x_1}(k, c, a) + p^{x_2}(k, c, a)$, $\forall k, c$, is not necessarily equal to one: it can be either zero or one. Transition rates $T(s, s', a)$ between states $s = (n^{x_1}, n^{x_2})$ and s' are then expressed as:

$$\begin{cases} \Lambda(k, c)\ p^{x_1}(k, c, a) & \text{if } s' = (n^{x_1} + e^{x_1}(k, c, i), n^{x_2}) \\ \Lambda(k, c)\ p^{x_2}(k, c, a) & \text{if } s' = (n^{x_1}, n^{x_2} + e^{x_2}(k, c, i)) \\ n^{x_1}(k, c, i)\ \mu_c^{x_1}(s) & \text{if } s' = (n^{x_1} - e^{x_1}(k, c, i), n^{x_2}) \\ n^{x_2}(k, c, i)\ \mu_c^{x_2}(s) & \text{if } s' = (n^{x_1}, n^{x_2} - e^{x_2}(k, c, i)) \\ 0 & \text{Otherwise} \end{cases} \quad (5)$$

where $e^x(k, c, i)$ is defined as a ($N_Z \times N_C \times N_I^x$)-tuple containing all zeros except for the $(k, c, i)^{th}$ element, that is equal to one, and new arrivals join RAT x with the i^{th} QoS parameters couple proposed by action a. Hence, for example, when a class c user in zone Z_k joins RAT x_1, with the i^{th} QoS parameters couple, the network moves to state $s' = (n^{x_1} + e^{x_1}(k, c, i), n^{x_2})$.

The state dynamics can equivalently be characterized by the state transition probabilities $p(s, s', a)$ of the embedded chain:

$$p(s, s', a) = T(s, s', a) \cdot\ \tau(s, a) \quad (6)$$

where $\tau(s, a)$ is the expected sojourn time for each state-action pair, defined as follows:

$$\Bigg\{ \sum_x \sum_k \sum_c [\Lambda(k, c) p^x(k, c, a) + \sum_i n^x(k, c, i) \mu_c^x(s)] \Bigg\}^{-1} \quad (7)$$

D. Network Reward

To formulate optimization objectives, let $r(s,a)$ denote the permanence reward earned by the network in state s, when action a is adopted. Unlike the impulsive reward, received upon transitions, the permanence reward represents the benefit and penalty continuously received by the network whilst in state s (*i.e.,* it is actually defined on a per unit time basis). In our work, we express $r(s,a)$ as the sum of a network utility $N(s,a)$ and a blocking term $B(s,a)$:

$$r(s,a) = N(s,a) + B(s,a) \tag{8}$$

The network utility is given by:

$$N(s,a) = \sum_x \sum_k \sum_c \sum_i n^x(k,c,i) d^x(k,c,i) \tag{9}$$

where $d^x(k,c,i)$ represents the average throughput of class c users in zone Z_k, that have joined RAT x with the i^{th} ($d_{min}(x)$, $d_{max}(x)$) couple. In fact, mobiles are first provided with their minimum guaranteed throughput given by $d_{min} \cdot g(k)$. Then, fair time scheduling is used to provide them with up to their maximum throughput given by $d_{max} \cdot g(k)$. Remaining resources may afterwards be equitably shared (*i.e.,* after receiving their maximum throughput, all mobiles have the same priority leading to fair time scheduling).

Furthermore, the blocking term reflects the penalty of rejecting future arrivals. $B(s,a)$ is thus proportional to the arrival rates in blocking states, and is expressed as follows:

$$B(s,a) = -\, b \cdot \sum_k \sum_c \Lambda(k,c)\left(1 - \sum_x p^x(k,c,a)\right) \tag{10}$$

where b is the cost per unit time inflicted on the network for blocking a new arrival.

E. Uniformization

In our work, we make use of the *Policy Iteration* algorithm to solve the SMDP problem (*i.e.,* to determine the action the network takes in each state). A stage of uniformization is thus required. The continuous-time Markov chain is transformed to its discrete equivalent.

Time is first discretized into intervals of constant duration τ, that is smaller than the expected sojourn time in any state: $0 \leq \tau < \tau(s,a)$, $\forall s \in \mathcal{S}$.

Transition probabilities are then modified as follows:

$$\begin{cases} \bar{p}(s,s',a) = p(s,s',a)\frac{\tau}{\tau(s,a)} & \text{for } s' \neq s \\ \bar{p}(s,s',a) = 1 - \sum_{s' \neq s} \bar{p}(s,s',a) & \text{Otherwise} \end{cases} \tag{11}$$

where $\bar{p}(s,s',a)$ represents the probability that the network moves from state s to s' within τ, when action a is adopted.

Moreover, the reward is also modified as follows: $\bar{r}(s,a) = r(s,a)\tau$, where $\bar{r}(s,a)$ is the reward earned for a time τ.

F. Policy Iteration Algorithm

A policy π is a mapping from $\mathcal{S}$ to $\mathcal{A}$. $\pi(s)$ represents the action to take in state s. Let $H_\pi(s) = s, s_1, s_2, ..., s_n, ...$ be a trajectory of the Markov chain, when policy π is adopted. The long-term discounted reward $dr(H_\pi(s))$ of state s is the discounted sum of the rewards earned on that trajectory (that starts from s), and is expressed as follows:

$$\bar{r}(s,\pi(s)) + \psi\bar{r}(s_1,\pi(s_1)) + ... + \psi^n\bar{r}(s_n,\pi(s_n)) + ...$$

where ψ is the discounting factor ($0 < \psi < 1$). In our work, we set the value function of state s, denoted by $V_\pi(s)$, as the expected value of $dr(H_\pi(s))$ over all possible trajectories.

Our goal is to find an optimal policy π_{opt}, that maximizes the expected long-term discounted reward of each state:

$$V_{\pi_{opt}}(s) \geq V_\pi(s), \qquad \forall s, \pi$$

We therefore use the following *Policy Iteration* algorithm:

- Step 0 (Initialization): We choose an arbitrary stationary policy π.
- Step 1 (Value Determination): Given the current policy π, we solve the following system of linear equations to calculate the discounted value function V_π of all states:

$$V_\pi(s) = \bar{r}(s,\pi(s)) + \psi \sum_{s' \in \mathcal{S}} \bar{p}(s,s',\pi(s)) V_\pi(s')$$

- Step 2 (Policy Improvement): When any improvement is possible, we update the current policy π. For each s $\in \mathcal{S}$, we find:

$$\hat{\pi}(s) = \arg \max_{a \in \mathcal{A}(s)} \left\{ \bar{r}(s,a) + \psi \sum_{s' \in \mathcal{S}} \bar{p}(s,s',a) V_\pi(s') \right\}$$

- Step 3 (Convergence test): If $\hat{\pi} = \pi$, the algorithm is stopped with $\pi_{opt} = \pi$. Otherwise, we go to step 1.

VI. Reinforcement learning

In the previous section, knowing $r(s,a)$ and $p(s,s',a)$, an optimal policy π_{opt} is solved through the *Policy Iteration* algorithm. The transition probability function $p(s,s',a)$ depends on user arrival and departure rates, needs, preferences, and decision-making algorithms. However, when $p(s,s',a)$ may not be easily obtained, reinforcement learning (RL) turns out to be a good fit to derive network information. The network does not estimate user behavior, but rather learns what action to take by trial-and-error. Among the different existing RL algorithms, we select Q-learning [35] for its simplicity. Although originally used to solve Markov decision processes, Q-learning may be applied with slight modifications to semi-Markov decision processes [36].

A. SMDP Q-learning Algorithm

The network interacts with its environment over a sequence of discrete time-steps $(t, t+1, t+2, ...)$, trying to learn what QoS parameters to signal. These time-steps refer to time intervals of fixed duration τ. The *quality* function of state-action pair $(s,\pi(s))$, denoted by $Q_\pi(s,\pi(s))$, is defined as the expected long-term discounted reward of state s, using policy π. Our aim is to find an optimal policy π_{opt}, that maximizes the *quality* function of each state s, also referred to as its Q-value:

$$\pi_{opt}(s) = \arg \max_{a \in \mathcal{A}(s)} Q_{\pi}(s,a), \qquad \forall s, \pi$$

Without knowledge of $p(s,s',a)$, the network, also referred to as the agent, iteratively learns optimal Q-values. At discrete time-steps, when the network state has changed, the network action terminates. QoS parameters to be signaled may have to vary. Unlike in Markov decision processes, where all actions are assumed to take constant time to complete, actions in our work can span several time-steps. They are said to be temporally-abstract. At time-step t, when state-action pair (s,a) is visited (*i.e.,* when the network in state s selects and performs action a), the network earns reward R, and ends in state s' at $t+k$. The Q-value of state-action pair (s,a) is then updated as follows:

$$Q(s,a) \leftarrow Q(s,a) + \rho\Big(R + \psi^k \max_{a' \in \mathcal{A}} \{Q(s',a')\} - Q(s,a)\Big) \tag{12}$$

where ρ is the learning rate ($0 < \rho < 1$), that determines to what extent the learned Q-value will override the old one. When $\rho = 0$, the network does not learn. When $\rho = 1$, the network considers only the most recent Q-value. R is the discounted accumulation of all single-step rewards r_τ, received while executing action a for a time τ, and is given by:

$$R = \sum_{i=0}^{k-1} \psi^i \; r_\tau$$

Moreover, it has been proved that, while the number of visits of each state-action-pair is sufficiently large, and ρ is reduced to zero over time, $Q(s,a)$ is guaranteed to converge to $Q_{\pi_{opt}}(s,a)$ [35].

B. Exploration and Exploitation

At decision epochs, the network decides, randomly or based on previously learned Q-values, what QoS parameters to signal. To receive high reward, the network may prefer actions it has tried in the past and found effective. This is known as the exploitation mode. Yet, to discover effective ones, the network needs to try actions it has not selected before. It may then randomly select one of the possible actions, aiming to enhance its future decisions. This is known as the exploration mode. Since Q-learning is an online iterative learning algorithm, exploration and exploitation should be simultaneously performed. The agent must discover a variety of actions, and progressively favor effective ones. However, to estimate reliable Q-values, actions need to be sufficiently tested.

In our work, we adopt an ϵ-greedy exploration-exploitation policy. At decision epochs, the network in state s explores with probability $\epsilon(s)$, and exploits stored Q-values with probability $1-\epsilon(s)$. To enhance long-term network performance, exploring is never stopped, but rather reduced over time. We define $\beta(s,a)$ to be the number of visits of state-action pair (s,a) up to current time-step, and choose $\epsilon(s)$ to be as follows:

$$\epsilon(s) = \frac{1}{ln(\sum_{a \in \mathcal{A}} \beta(s,a) + 3)} \tag{13}$$

$\epsilon(s)$ then belongs to $[0,1]$, and has a logarithmic decay. Furthermore, for $Q(s,a)$ to converge to optimal Q-values, we set ρ to be a state-action pair varying over time:

$$\rho(s,a) = \frac{1}{\sqrt{\beta(s,a) + 3}}$$

Algorithm 1 describes our SMDP Q-learning algorithm for deriving network information. We summarize below the main steps. Q-values are first set to zero. The network state is randomly initialized. Once in state s, depending on $\epsilon(s)$, exploration or exploitation is executed. In exploration mode, the network randomly selects and performs action a. However, in exploitation mode, it opts for the action with the maximum Q-value: $a = \max_a Q(s,a)$. After, at each time-step, the network state is observed. While the network is in state s, action a is maintained, and the discounted accumulation of single-step rewards R is updated. Yet, if it is in state s' (*i.e.,* the network state has changed), action a is terminated, and $Q(s,a)$ is updated according to equation 12. This is repeated until the end of the learning period.

Initialize
- Q-values: $Q(s,a) \leftarrow 0, \forall s \in \mathcal{S}$ and $a \in \mathcal{A}$
- Number of state-action visits: $\beta(s,a) \leftarrow 0, \forall s \in \mathcal{S}$ and $a \in \mathcal{A}$
- Time-step: $t \leftarrow 0$

repeat
- Observe state s
- **if** *exploration* **then**
  - choose action a at random
- **else**
  - choose $a = \max_a Q(s,a)$
- **end**
- $\beta(s,a) \leftarrow \beta(s,a) + 1$
- Update $\epsilon(s)$ according to equation 13
- $R \leftarrow 0$
- $k \leftarrow 0$
- **while** *the network is in state* s **do**
  - Perform a
  - Wait for a fixed duration τ
  - Observe reward r_τ
  - $R \leftarrow R + \psi^k \; r_\tau$
  - $k \leftarrow k + 1$
- **end**
- Observe state s'
- Update $Q(s,a)$ according to equation 12
- $s \leftarrow s'$
- $t \leftarrow t + k$

until *End of the learning period*;

Algorithm 1: SMDP Q-learning

VII. Performance Results

For illustration, we consider a heterogeneous cellular network composed of mobile WiMAX and LTE, respectively designated by W and L. For simplicity, users are of two types: those with good radio conditions (or cell-center users that belong to zone 1) and those with bad radio conditions (or cell-edge users that belong to zone 2). Their peak throughputs, when connected alone to mobile WiMAX and LTE cells, are depicted in Table I. Further, class c arrivals are assumed to be uniformly distributed over the two zones, and to follow a Poisson process of rate $\Lambda_c = \Lambda$ (*i.e.,* $\Lambda(k,c) = \Lambda/N_Z$, $\forall k, c$).

RAT	k = 1	k = 2
Mobile WiMAX (3 MHz)	9.9 Mb/s	4.4 Mb/s
LTE (5 MHz)	16.6 Mb/s	7.4 Mb/s

TABLE I
Peak throughputs in Mobile WiMAX and LTE

Moreover, for streaming sessions, we suppose that $R_{av} = 1$ Mb/s, $R_{max} = 1.5$ Mb/s, and $1/\mu_1 = 45$ s. For elastic sessions, we consider that $L = 5$ Mbytes, and R_C is fixed to either 1.25 or 0.75 Mb/s, depending on the QoS-maximizing or cost-minimizing preferences of mobile users. For network information, we assume that $cost(W) = 4$, $cost(L) = 6$, $N_I^W = N_I^L = 3$, $I^W = \{(0,0),(0.5,1),(1,1.5)\}$ Mb/s, and $I^L = \{(0,0),(0.75,1.25),(1.5,2)\}$ Mb/s.

The probabilities $p^x(k,c,a)$ are calculated according to the satisfaction-based multi-criteria decision-making method, we have introduced in [14]. They mainly depend on user preferences, traffic classes and throughput demands. Note that half of the users are ready to pay for better performances.

For comparison purposes, we also investigate the staircase tuning policy [23]. Load factors are defined as the ratios of the number of guaranteed allocated RUs to the total number of RUs. The highest QoS parameters are first signaled. Next, when a RAT load factor exceeds S_1 threshold, QoS parameters are reduced following a step function (cf. Fig. 3). However, if S_2 is reached, QoS incentives are no longer provided. QoS parameters to signal in RAT x, depending on the load factor ϕ^x, are reported in table II.

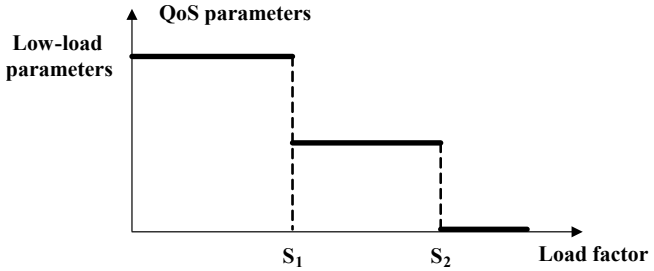

Fig. 3. QoS parameters reduction using the staircase policy

Before we discuss performance results, we remind in table III some notations, useful for the following.

A. Staircase Policy

Using the staircase policy, we study the impact of S_1 and S_2 thresholds on network performance. Figures 4 and 5 respectively show the network throughput, defined as the sum of user throughputs, and the blocking probability, as a function of the cell arrival rate Λ, for different threshold values. For fixed S_1, the higher S_2 the more mobiles are admitted. Yet, higher S_2 thresholds limit user throughputs to their guaranteed ones. Besides, for fixed S_2, the lower S_1 the less mobiles benefit from the largest QoS guarantees, but much more are admitted with reduced QoS parameters. Therefore, the average number of simultaneous sessions increases.

QoS parameters	$\phi^x < S_1$	$S_1 \leq \phi^x \leq S_2$	$\phi^x > S_2$
$d_{min}(W)$	1 Mb/s	0.5 Mb/s	0
$d_{max}(W)$	1.5 Mb/s	1 Mb/s	0
$d_{min}(L)$	1.5 Mb/s	0.75 Mb/s	0
$d_{max}(L)$	2 Mb/s	1.25 Mb/s	0

TABLE II
QoS parameters depending on the load factor ϕ^x

Parameters	Notation
Tuning thresholds of the staircase policy	S_1, S_2
Discount factor	ψ
Cell arrival rate	Λ
Blocking cost	b
Blocking term (penalty term)	B
Duration of learning periods	T
Duration of time-steps	τ

TABLE III
Summary of notations

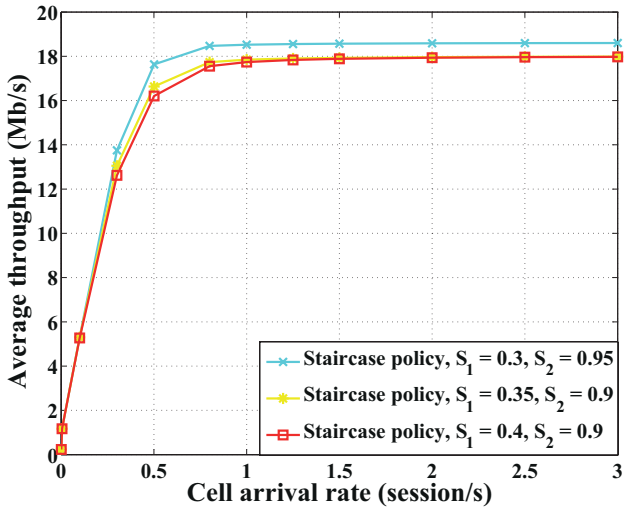

Fig. 4. Staircase policies: network throughput

Obviously, the more mobiles are admitted for a fixed cell arrival rate (*i.e.,* the lower S_1 or the higher S_2), the lower the blocking probability. Also, the network throughput augments. Typically, streaming sessions have limited throughput demands, and hence the more mobiles are admitted the larger the network throughput will potentially be.

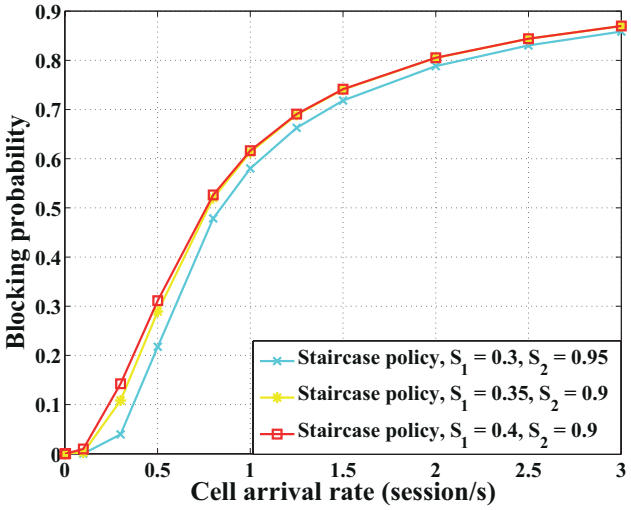

Fig. 5. Staircase policies: blocking probability

B. Optimal Policy

The optimal policy, solved through the *Policy Iteration* algorithm, and the staircase policy are compared. Using the optimal policy, we study the impact of the blocking cost b, and the discount factor ψ on network performance.

1) Impact of the blocking cost: We start by inspecting the impact of the blocking cost b on network performance. So as to enlarge the number of states involved in the value function, the discount factor ψ is fixed at 0.99.

Figure 6 illustrates the average reward as a function of the cell arrival rate Λ, for different blocking costs. When b is zero, the reward function is reduced to the network utility representing the sum of user throughputs. Otherwise, it also includes a penalty term, that is proportional to the blocking cost b and to the cell arrival rate.

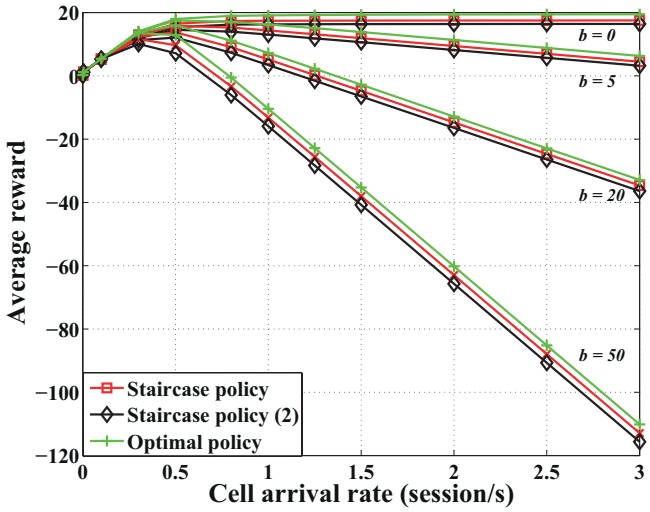

Fig. 6. Impact of b on network reward

At low arrival rate, no blocking occurs leading to similar rewards regardless of b. The reward function, reduced to the network throughput, then increases with the cell arrival rate. Yet, as the latter increases further, or equivalently, when the average number of simultaneous sessions augments, network resources are always nearly exhausted, and not enough are left to cope with future arrivals. Therefore, the blocking probability (*i.e.,* the long-term fraction of time spent in blocking states) also increases. Moreover, and since the penalty term is proportional to the cell arrival rate, the reward function received by the network whilst in a blocking state is as reduced as the arrival rate is increased. For all these reasons, the average reward decreases more when the cell arrival rate increases, except for b equals zero. In fact, when b is null, the average reward stagnates at high arrival rate. It represents the long-term sum of user throughputs. Otherwise, the average reward obviously decreases with increasing blocking costs. We further note that the optimal policy always outperforms the staircase one. However, when S_1 and S_2 are respectively set to 0.3 and 0.95, the staircase policy provides higher network reward in comparison with the case when $S_1 = 0.35$ and $S_2 = 0.85$, denoted as Staircase policy (2).

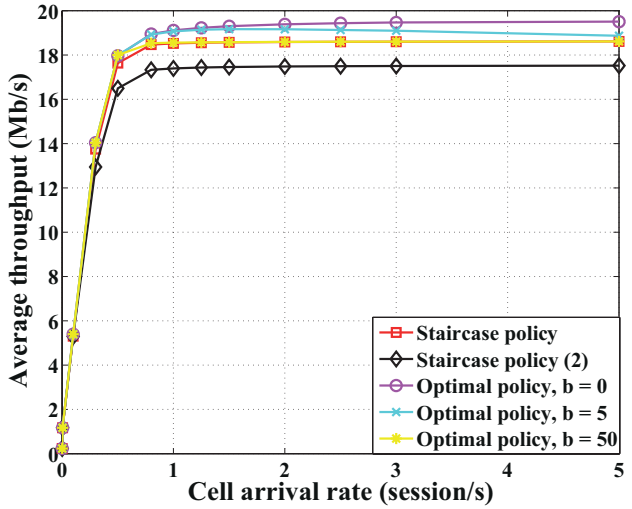

Fig. 7. Impact of b on network throughput

Besides, the impact of the penalty term on the reward function, and thereafter on the optimal policy, strongly depends on the blocking cost b. On the one hand, the higher b the more the network avoids blocking actions, even if at the expense of the network utility. On the other hand, the lower b, the more the network tries to maximize its throughput, even if leading to more blocking states. We, respectively, depict in figures 7 and 8 the network throughput and the percentage in number of blocking states, as a function of the cell arrival rate. The optimal policy is illustrated for different values of b. Particularly, when b is zero, the network throughput, but also the percentage of blocking states, are maximized. The blocking cost b may therefore be tuned to control optimization objectives. Further, when $S_1 = 0.3$ and $S_2 = 0.95$, the staircase policy achieves a higher throughput in comparison with when S_1 and S_2 are respectively set to 0.35 and 0.85. As a matter of fact, when these thresholds are carefully chosen,

the staircase policy provides quite similar performances as the optimal one ($b = 50$). They both effectively avoid blocking actions and guide user decisions. In the remaining, we only consider the case where $S_1 = 0.3$ and $S_2 = 0.95$.

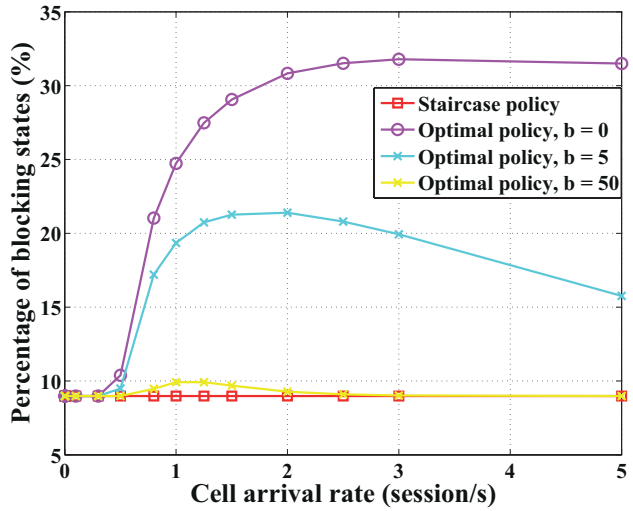

Fig. 8. Impact of b on the percentage of blocking states

It is worth noting that for a given b, when the cell arrival rate is different, the state dynamics and penalty terms are also different. This may lead to dissimilar optimal policies. Thus, and as shown in Fig. 8, the percentage in number of blocking states first increases with the cell arrival rate. Then, when the latter increases further, for b different from zero, this percentage decreases as the penalty term becomes relatively very significant.

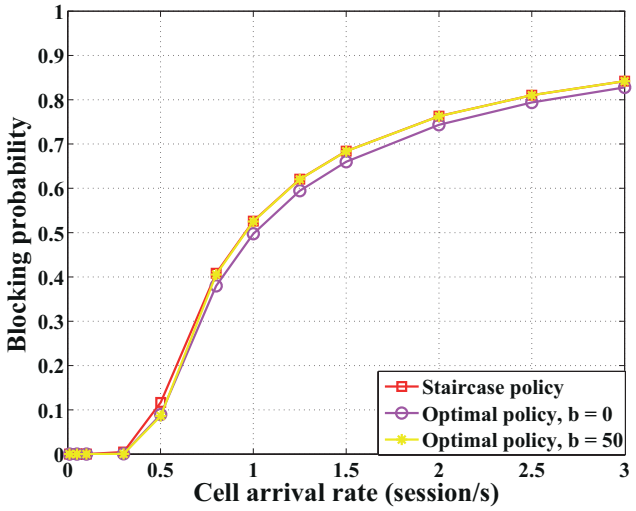

Fig. 9. Impact of b on blocking probability for elastic sessions

Moreover, the blocking probability $\mathbb{P}_b$ depends not only on the number of blocking states, but mostly on the stationary distribution achieved by the different policies (*i.e.,* on the long-term fraction of time spent in the different states). In the following, to efficiently analyze the impact of the blocking cost on $\mathbb{P}_b$, we separately consider streaming and elastic sessions.

The service time of elastic sessions depends both on their size assumed to be exponentially distributed with a mean of 5 Mbytes, and on their perceived throughputs. As shown before, the lower b, the higher the network throughput leading to lower average service times. When the optimal policy is adopted (*i.e.,* the actions are fixed to the optimal ones), the SMDP may be reduced to a Markov chain, where departure rates increase with decreasing blocking costs. As a result, for a given cell arrival rate, the lower b, the lower the long-term number of simultaneous sessions. This also means that, although the lower b the higher the percentage of blocking states, the long-term fraction of time spent in these states is reduced as b is low. Accordingly, the lower b, the lower $\mathbb{P}_b$ for elastic sessions as illustrated in Fig. 9.

Nevertheless, the service time of streaming sessions exclusively depends on their duration, considered to be exponentially distributed with a mean of 45 s. Thereby, maximizing the network throughput will not reduce their average service times. Consequently, as the number of blocking states increases with decreasing b, the blocking probability for streaming sessions also increases (cf. Fig. 10). The long-term fraction of time spent in blocking states will actually be higher. Here again, for both traffic classes, the performance of the staircase policy, with carefully chosen S_1 and S_2 thresholds, is comparable to the optimal solution ($b = 50$).

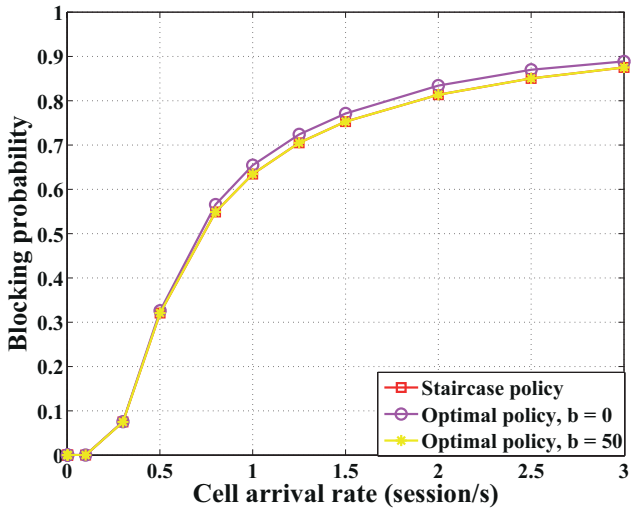

Fig. 10. Impact of b on blocking probability for streaming sessions

2) Impact of the discount factor: In this paragraph, we investigate the impact of the discount factor ψ on network performance. When the blocking cost b is set to zero, the network reward is reduced to the sum of user throughputs.

Figures 11 and 12 respectively illustrate the network throughput and the blocking probability as a function of the cell arrival rate, for different ψ values. Recall that the higher ψ, the larger the number of states involved in the value function. Also, next states contribute more to the expected long-term network reward as ψ gets higher.

The discount factor ψ can thus be tuned to control the optimization scope. Typically, higher ψ values imply more long-run optimization, leading to higher throughput and lower blocking probability.

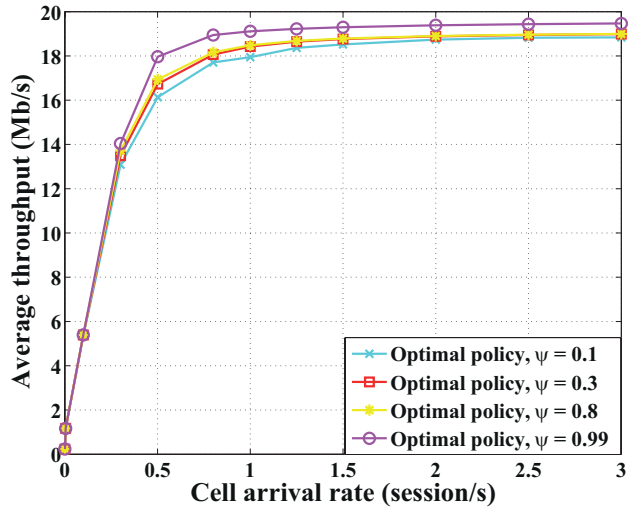

Fig. 11. Impact of ψ on network throughput

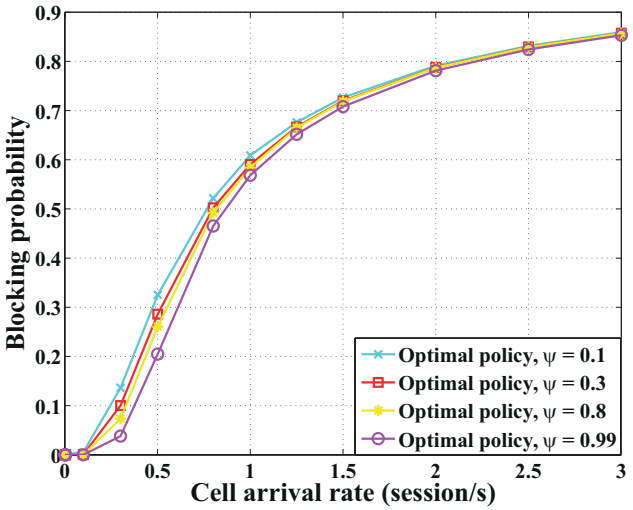

Fig. 12. Impact of ψ on blocking probability

Further, we note that the network throughput at low arrival rate and the blocking probability at high arrival rate are obviously quite similar, regardless of the discount factor.

Figures 13 and 14 compare the optimal policy with the staircase one. On the one hand, we notice that, at low arrival rate (typically below 1), the staircase policy outperforms the optimal one with $\psi = 0.3$ and $\psi = 0.8$. This means that the intuitive and low-complexity staircase policy efficiently guides user decisions at low arrival rate. Yet, to maximize network performance, the number of states that are involved in the value function should be large enough. This can be seen with $\psi = 0.99$.

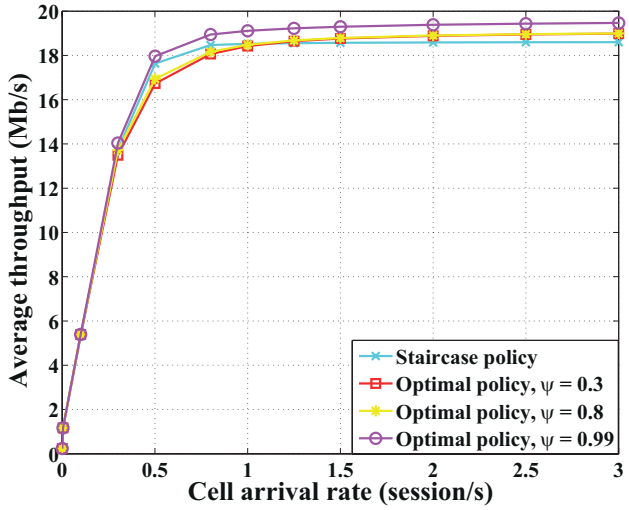

Fig. 13. Optimal vs. staircase policies: network throughput

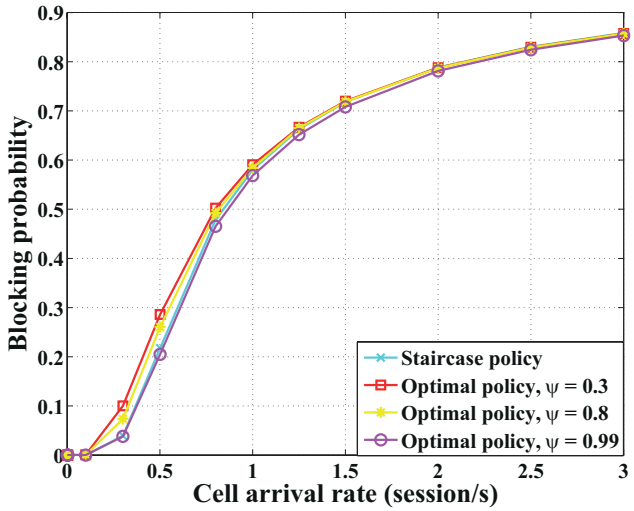

Fig. 14. Optimal vs. staircase policies: blocking probability

On the other hand, when the cell arrival rate increases, taking into account next states becomes more relevant. In fact, when the network is expected to approach its saturation, deriving QoS parameters considering future arrivals enhances long-term network performance. Also, reducing QoS parameters in all serving RATs, following the staircase policy, proves to provide close performance to the optimal solution (cf. Fig. 13 and 14).

C. Learning-based Policy

In what follows, the learning-based policy ($\psi = 0.99$), the optimal policy ($b = 0, \psi = 0.99$) and the staircase policy are compared. Using the Q-learning algorithm, the agent interacts with its environment over a sequence of $T = 100000$ and $T = 250000$ time-steps, of fixed duration $\tau = 0.5$ s. Performance metrics are then averaged over 20 learning periods.

Figures 15 and 16 respectively show the network throughput and the blocking probability, as a function of the cell arrival rate. The optimal solution, solved using the *Policy Iteration* algorithm, provides an upper bound on the network throughput. It also brings the lowest blocking probability, and consequently the best network performance. However, the optimal policy suffers from high computational complexity. For a fixed discount factor, the *Policy Iteration* algorithm is shown to run in at most $\frac{N_s^2(N_a-1)}{1-\psi}\cdot \log(\frac{N_s^2}{1-\psi})$ iterations, where N_s is the number of states, N_a the number of actions, and ψ the fixed discount factor [37].

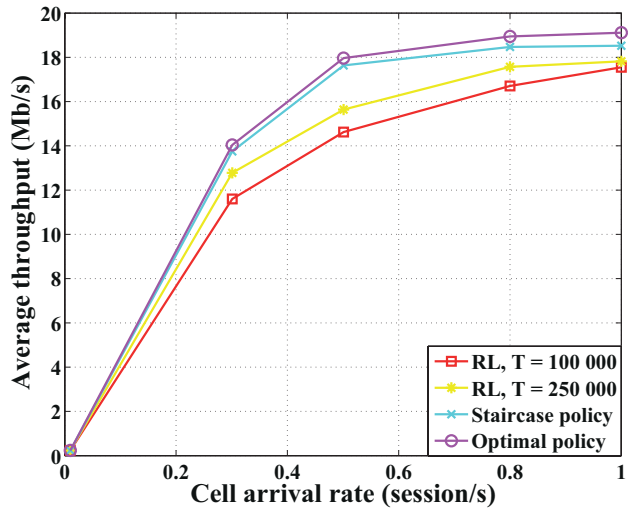

Fig. 15. RL-based vs. optimal vs. staircase policies: network throughput

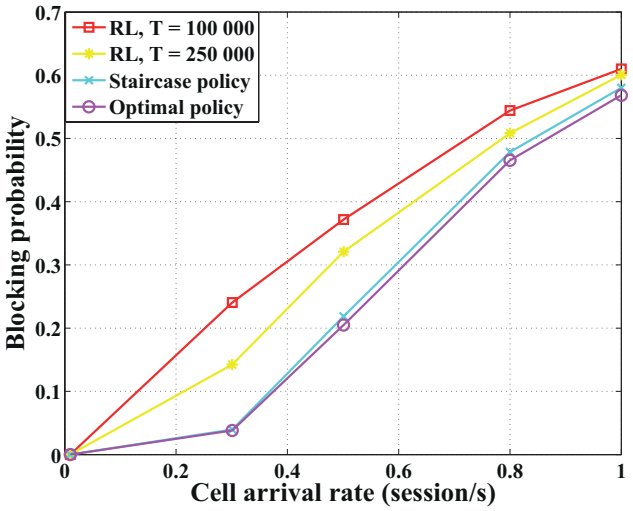

Fig. 16. RL-based vs. optimal vs. staircase policies: blocking probability

Moreover, and as discussed before, the staircase policy provides very close performance to the optimal solution despite its low complexity. Yet, a practical challenge is to efficiently set S_1 and S_2 values. When our heuristic requires no knowledge on network parameters, its performance strongly depends on the choice of the tuning thresholds.

Furthermore, unlike the optimal and the heuristic solutions, the learning-based one needs no parameterization. Theoretically, after an infinite learning period, our Q-learning algorithm converges to the optimal solution. In our work, we stop learning after a realistic duration of $T = 100000$ and $T = 250000$ time-steps. Better performances are obviously observed when $T = 250000$, in comparison with when $T = 100000$. Yet, when learning periods are voluntary limited, both the optimal and the heuristic policies outperform the learning-based ones.

VIII. Conclusion

In this paper, we have addressed the radio access technology selection in heterogeneous cellular networks. We have proposed a network-assisted approach, aiming to jointly enhance network performance and user experience. As a matter of fact, the network provides information for the mobiles to make decisions regarding selection of their most appropriate RAT. Deriving network information was formulated as a semi-Markov decision process, and optimal policies were solved through the *Policy Iteration* algorithm. However, a heavy computational load was required to find optimal solutions. Further, as network parameters may not be easily obtained, we have introduced a RL-based algorithm to determine what to signal to mobiles. Although our learning-based algorithm requires no prior parameterization, it does not react fast to sudden parameter changes. Moreover, we have investigated the staircase policy, and proved its efficiency for implementation in practice. When tuning thresholds are pertinently chosen, our heuristic achieves close performance to the optimal solution.

Appendix
User Decision-Making

We consider a mobile user, in zone Z_k, that needs to join one of the serving RATs. For RAT x, the network broadcasts the three parameters $d_{min}(x)$, $d_{max}(x)$, and $cost(x)$. From the user point of view, these parameters are the values of the decision criteria, used to evalute available RATs. As $d_{min}(x)$ and $d_{max}(x)$ are derived for the most robust modulation and coding scheme, the mobile has first to determine its perceived QoS parameters, as signaled by the network: $d_{min}(x)\cdot g(k)$ and $d_{max}(x)\cdot g(k)$. Second, using the satisfaction-based multi-criteria decision-making method we have introduced in [14], the mobile computes a utility function for each of the available RATs, and selects the one with the highest score.

The utility function of a class c user, for RAT x, is defined as follows:

$$U^c(x) = w^c_{d_{min}}\cdot \hat{d}^c_{min}(x) + w^c_{d_{max}}\cdot \hat{d}^c_{max}(x) + w^c_{cost}\cdot \widehat{cost}^{\,c}(x)$$

where $\hat{d}^c_{min}(x)$, $\hat{d}^c_{max}(x)$, and $\widehat{cost}^{\,c}(x)$ are respectively the normalized values of $d_{min}(x)$, $d_{max}(x)$, and $cost(x)$. $w^c_{d_{min}}$, $w^c_{d_{max}}$, and w^c_{cost} are the weights of the decision criteria, reflecting user preferences.

The normalizing functions depend on user traffic class c, and throughput demands. As streaming sessions ($c = S$) are usually characterized by a minimum, an average and a maximum

bandwidth requirement, their throughput normalizing function is modeled as an sigmoid function:

$$\hat{d'}^{S}(x) = 1 - \exp(\frac{-\alpha(\frac{d'(x).g(k)}{R_{av}})^2}{\beta + (\frac{d'(x).g(k)}{R_{av}})}) \quad (14)$$

where $d' = \{d_{min}, d_{max}\}$. R_{av} represents session needs: an average throughput demand. α and β are two positive constants that determine the shape of the sigmoid function.

Also, when elastic sessions ($c = E$) adapt to resource availability, requiring no QoS guarantees, $d_{min}(x)$ is completely ignored (*i.e.,* $w^E_{d_{max}} = 0$). Moreover, their throughput normalizing function for d_{max} has a concave shape. $\hat{d}^E_{max}(x)$ increases slowly as the throughput exceeds the comfort throughput demand R_c of the user (*i.e.,* the average throughput beyond which, user satisfaction exceeds 63% of maximum satisfaction).

$$\hat{d}^E_{max}(x) = 1 - \exp(-\frac{d_{max}(x).g(k)}{R_c}) \quad (15)$$

However, the cost normalizing function is modeled as a Z-shaped function for both traffic classes. $\widehat{cost}^{c}(x)$ then decreases rapidly with the cost.

$$\widehat{cost}^{c}(x) = \exp(-\frac{cost(x)^2}{\lambda^c}), c \in \{S, E\} \quad (16)$$

where λ^c represents the cost tolerance parameter, a positive constant that determines the shape of the Z-shaped function.

Depending on user preferences (*e.g.,* QoS-maximizing or cost-minimizing preferences), and needs (*i.e.,* traffic class), the cost tolerance parameter λ and the weights of the decision criteria are reported in table IV.

Preferences	**Traffic class**	λ	$w_{d_{min}}$	$w_{d_{max}}$	w_{cost}
QoS-max	Streaming	60	14/30	7/30	0.3
QoS-max	Elastic	60	0	0.7	0.3
Cost-min	Streaming	25	0.2	0.1	0.7
Cost-min	Elastic	25	0	0.3	0.7

TABLE IV
COST TOLERANCE PARAMETERS AND DECISION CRITERIA WEIGHTS

REFERENCES

[1] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013–2018*, Cisco, February 2014.

[2] M. Nicosia, R. Klemann, K. Griffin, S. Taylor, B. Demuth, J. Defour, R. Medcalf, T. Renger, and P. Datta, "Rethinking Flat Rate Pricing for Broadband Services." Cisco, July 2012, White Paper.

[3] C. Cicconetti, "5G Radio Network Architecture," March 2013, White Paper.

[4] 3GPP TS 32.521, "Telecommunication Management; Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Requirements," March 2010.

[5] M. L. Puterman, *Markov Decision Processes*. John Wiley, 1994.

[6] E. Aryafar, A. Keshavarz-Haddad, M. Wang, and M. Chiang, "RAT Selection Games in HetNets," in *Proc. IEEE Conference on Computer Communications (INFOCOM)*, April 2013.

[7] K. Khawam, M. Ibrahim, J. Cohen, S. Lahoud, and S. Tohme, "Individual vs. Global Radio Resource Management in a Hybrid Broadband Network," in *Proc. IEEE International Conference on Communications (ICC)*, June 2011.

[8] M. Ibrahim, K. Khawam, and S. Tohme, "Congestion Games for Distributed Radio Access Selection in Broadband Networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, December 2010.

[9] D. Niyato and E. Hossain, "Dynamics of Network Selection in Heterogeneous Wireless Networks: An Evolutionary Game Approach," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 2008 – 2017, May 2009.

[10] P. Coucheney, C. Touati, and B. Gaujal, "Fair and Efficient User-Network Association Algorithm for Multi-Technology Wireless Networks," in *Proc. IEEE Conference on Computer Communications (INFOCOM)*, April 2009.

[11] O. Ercetin, "Association Games in IEEE 802.11 Wireless Local Area Networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 5136 – 5143, December 2008.

[12] D. Kumar, E. Altman, and J.-M. Kelif, "User-Network Association in a WLAN-UMTS Hybrid Cell: Global & Individual Optimality," INRIA, Rapport de recherche RR-5961, 2006.

[13] Q.-T. Nguyen-Vuong, N. Agoulmine, E. Cherkaoui, and L. Toni, "Multicriteria Optimization of Access Selection to Improve the Quality of Experience in Heterogeneous Wireless Access Networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1785 – 1800, 2013.

[14] M. E. Helou, S. Lahoud, M. Ibrahim, and K. Khawam, "Satisfaction-based Radio Access Technology Selection in Heterogeneous Wireless Networks," in *Proc. IEEE IFIP Wireless Days Conference (WD)*, November 2013.

[15] M. E. Helou, M. Ibrahim, S. Lahoud, and K. Khawam, "Radio Access Selection Approaches in Heterogeneous Wireless Networks," in *Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, October 2013.

[16] I. Chamodrakas and D. Martakos, "A Utility-Based Fuzzy TOPSIS Method for Energy Efficient Network Selection in Heterogeneous Wireless Networks," *Applied Soft Computing*, vol. 12, no. 7, pp. 1929 – 1938, July 2012.

[17] F. Bari and V. C. Leung, "Automated Network Selection in a Heterogeneous Wireless Network Environment," *IEEE Networks*, vol. 21, no. 1, pp. 34 – 40, January 2007.

[18] E. Stevens-Navarro and V. Wong, "Comparison Between Vertical Handoff Decision Algorithms for Heterogeneous Wireless Networks," in *IEEE Vehicular Technology Conference (VTC Spring)*, May 2006.

[19] Q. Song and A. Jamalipour, "Network Selection in an Integrated Wireless LAN and UMTS Environment Using Mathematical Modeling and Computing Techniques," *IEEE Wireless Communications*, vol. 12, no. 3, pp. 42 – 48, June 2005.

[20] ——, "A Network Selection Mechanism for Next Generation Networks," in *Proc. IEEE International Conference on Communications (ICC)*, May 2005.

[21] W. Zhang, "Handover Decision Using Fuzzy MADM in Heterogeneous Networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, March 2004.

[22] O. Falowo and H. Chan, "RAT Selection for Multiple Calls in Heterogeneous Wireless Networks Using Modified TOPSIS Group Decision-Making Technique," in *Proc. IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, September 2011.

[23] M. E. Helou, S. Lahoud, M. Ibrahim, and K. Khawam, "A Hybrid Approach for Radio Access Technology Selection in Heterogeneous Wireless Networks," in *Proc. European Wireless Conference (EW)*, April 2013.

[24] L. Zhu, F. Yu, B. Ning, and T. Tang, "Cross-Layer Handoff Design in MIMO-Enabled WLANs for Communication-Based Train Control (CBTC) Systems," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 4, pp. 719 – 728, 2012.

[25] L. Zhu, F. R. Yu, B. Ning, and T. Tang, "Handoff Management in Communication-Based Train Control Networks Using Stream Control Transmission Protocol and IEEE 802.11p WLANs," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 211 – 226, 2012.

[26] X. Zhang, H. Jin, X. Ji, Y. Li, and M. Peng, "A separate-SMDP Approximation Technique for RRM in Heterogeneous Wireless Networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, April 2012.

[27] J. P. Singh, T. Alpcan, P. Agrawal, and V. Sharma, "A Markov Decision Process based Flow Assignment Framework for Heterogeneous Network Access," *Wireless Network*, vol. 16, no. 2, pp. 481 – 495, February 2010.

[28] M. Ibrahim, K. Khawam, and S. Tohme, "Network-Centric Joint Radio Resource Policy in Heterogeneous WiMAX-UMTS Networks for

Streaming and Elastic traffic," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, April 2009.
[29] M. Coupechoux, J.-M. Kelif, and P. Godlewski, "Network controlled joint radio resource management for heterogeneous networks," in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, May 2008.
[30] ——, "SMDP Approach for JRRM Analysis in Heterogeneous Networks," in *Proc. European Wireless Conference (EW)*, June 2008.
[31] H. Tabrizi, G. Farhadi, and J. Cioffi, "Dynamic Handoff Decision in Heterogeneous Wireless Systems: Q-learning Approach," in *Proc. IEEE International Conference on Communications (ICC)*, June 2012.
[32] C. Dhahri and T. Ohtsuki, "Q-learning Cell Selection for Femtocell Networks: Single- and Multi-user Case," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, December 2012.
[33] ——, "Learning-Based Cell Selection Method for Femtocell Networks," in *Proc. IEEE Vehicular Technology Conference (VTC Spring)*, May 2012.
[34] "IEEE Standard for Architectural Building Blocks Enabling Network-Device Distributed Decision Making for Optimized Radio Resource Usage in Heterogeneous Wireless Access Networks," *IEEE Std 1900.4-2009*, 2009.
[35] C. Watkins and P. Dayan, "Technical Note: Q-Learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.
[36] M. R. K. Ryan, "Hierarchical Reinforcement Learning: A Hybrid Approach," 2002.
[37] Y. Ye, "The Simplex and Policy-Iteration Methods Are Strongly Polynomial for the Markov Decision Problem with a Fixed Discount Rate," *Mathematics of Operations Research*, vol. 36, no. 4, pp. 593 – 603, November 2011.

Melhem El Helou (S'08–M'15) received the engineering and master degrees in communications and networking from the Ecole Supérieure d'Ingénieurs de Beyrouth (ESIB), Faculty of Engineering at Saint Joseph University of Beirut, Beirut, Lebanon, in 2009 and 2010, respectively and the PhD degree in communication networks from IRISA research institute, University of Rennes 1, Rennes, France and Saint Joseph University of Beirut, in 2014. He joined ESIB in September 2013 where he is currently an Assistant Professor (*fr:* Maître de conférences). His research interests include wireless communications, cellular technologies, radio resource management, and optimization of heterogeneous networks.

Marc Ibrahim is an assistant professor at Saint Joseph University of Beirut, Beirut, Lebanon and the director of CIMTI (Centre d'Informatique, de Modélisation, et de Technologies de l'Information). He got his engineering degree from the Faculty of Engineering at Saint Joseph University of Beirut in 2002, then his Master's degree from the same faculty in 2004. In 2009, he received his PhD in communication networks from the University of Versailles, France. His research activities orbit wireless networks and particularly focus on radio resource management and network selection as well as performance modeling and networks measurement.

Samer Lahoud received a Ph.D. in communication networks from Telecom Bretagne, Rennes. After his Ph.D. he spent one year at Alcatel-Lucent Bell Labs Europe. In 2007, he joined the University of Rennes 1 as an assistant professor. His research activities at the IRISA laboratory in Rennes focus on routing and resource allocation algorithms for wired and wireless communication networks. He has co-authored 60+ papers published in international journals and conference proceedings.

Kinda Khawam got her engineering degree from Ecole Supérieure d'Ingénieurs de Beyrouth (ESIB) in 2002, the Master's degree in computer networks from Telecom ParisTech, Paris, France in 2003, and the Ph.D. from the same school in 2006. She was a post-doctoral fellow researcher in France Telecom, Issy-Les-Moulineau, France in 2007. Currently, she is an associate professor and researcher at the University of Versailles in France. Her research interests include radio and energy resource management.

Dany Mezher is an associate professor of computer engineering at the Saint Joseph University of Beirut, Lebanon and acting chairperson of the Electrical Engineering Department in the Faculty of Engineering. He received a computer engineering degree from Saint Joseph University of Beirut in 1992 followed by a PhD degree from IRISA (France) in the field of High Performance Computing, in 2001. He was a visiting researcher at IRISA in 2001, 2002, 2004 and 2005. His research interests include High Performance Computing for numerical problems (namely eigenvalue problems), numerical simulations, Big Data, and software architecture.

Bernard Cousin (M'07) is a Professor of Computer Science at the University of Rennes 1, France, since 1992. Bernard Cousin received, in 1987, his PhD degree in computer science from the University of Paris 6. Between 1985 and 1992, he has been successively Assistant Professor at University of Paris 6 and Associate Professor at University of Bordeaux 1.

Currently, he is at the head of a research group on Advanced Networking. He is member of IRISA (a CNRS-University joint research laboratory located at Rennes in France). He has co-authored 150+ papers published in international journals and conferences.

His research interests include next generation Internet, green networking, all-optical networks, wireless networks, dependable networking, high speed networks, traffic engineering, multicast routing, network QoS management, network security and multimedia distributed applications.