Machine Learning Aided Cognitive RAT Selection for 5G Heterogeneous Networks

Juan S. Pérez, Sudharman K. Jayaweera
Communications and Information Sciences Laboratory
Department of Electrical and Computer Engineering
University of New Mexico, Albuquerque, NM, USA.
Emails: {jsperez, jayaweera}@unm.edu

Steven Lane
Air Force Research Laboratory
Kirtland Air Force Base
Albuquerque, NM, USA.

***Abstract*—The starring role of the Heterogeneous Networks (HetNet) strategy as the key Radio Access Network (RAN) architecture for future 5G networks poses serious challenges to the current user association (cell selection) mechanisms used in cellular networks. The max-SINR algorithm, although historically effective for performing this function, is inefficient at best and obsolete at worst in 5G HetNets. The foreseen embarrassment of riches and diversified propagation characteristics of network attachment points spanning multiple Radio Access Technologies (RAT) requires novel and creative context-aware system designs that optimize the association and routing decisions in the context of single-RAT and multi-RAT connections, respectively. This paper proposes a framework under these guidelines that relies on Machine Learning techniques at the terminal device level for Cognitive RAT Selection and presents simulation results to suppport it.**

I. Introduction

Currently, the academic and technical communities are collaborating to set the stage for future 5G networks. The vast amount of related literature foresees advancements that may lead to enhanced network capacity and decreased delay, new ways of spectrum access and sharing, the applicability of virtualization and software defined networks (SDN) concepts, and the development of new communication standards (e.g., mmWave) and Massive MIMO techniques [1], [2]. Further, there is consensus that the Heterogeneous Networks (HetNet) architecture will be the driving architecture for 5G [2].

Most mobile terminals today support the use of multiple Radio Access Technologies (RATs) (e.g., LTE, WiMAX, Wi-Fi, Bluetooth) through the use of several network interfaces or software defined radio (SDR) technology. In 5G HetNets, the availability of network attachment points through diversified RATs will continue to mature. This development will turn the classic user association problem into a complex decision process in order to guarantee efficacy and efficiency of the HetNet architecture. In this paper, we propose a framework for tackling the problem of determining which RAT standard and spectrum to utilize and which BS(s) or users to associate within the context of 5G HetNets.

The conventional mechanism for user association in cellular networks is based on the max-SINR rule: The user terminal connects to the BS that will result in the highest downlink SINR (signal to interference plus noise ratio). However, the max-SINR mechanism has proven to be insufficient in the context of HetNets [3], [4]. When the architecture of a network includes macro- and small-cells (i.e., micro- pico-, femto-cells), the reduced coverage of the latter makes them less atractive to the terminal devices regardless of how loaded the macro-cells might be, particularly when both network tiers are using the same spectrum.

Multi-parametric optimization solutions have been proposed in the literature to overcome the limitations of the max-SINR RAT selection approach by combining it with other criteria (e.g., cell load). However, the computational complexity of these approaches have made the task very difficult [4]. Furthermore, the associated problem of efficient and practical multi-parametric modeling (representation of the network state) has not been completely solved [5], [6]. The need for integrating multiple parameters in the optimization of network functionalities seems to be a design requirement for future networks. There is increasing agreement to look at these parameters under the umbrella of context-awareness [2], [3], [6], [7].

An ideal solution for the user association problem for HetNets is found at the intersection of three perspectives that reflect the approaches of several researchers: (1) A load balancing perspective (e.g., [8], [9]), with the main goal of improving the experience of both macro-cell users and offloaded users. (2) An enhanced mobility perspective (e.g., [5]), with the objective of making the mobility as seamless and transparent as possible to the user, and (3) A Self Organizing Network (i.e., distributed autonomous intelligence) perspective (e.g., [6], [10]), in order to reduce the overhead, increase the network efficiency and ease the management tasks for highly diverse and complex networks.

We propose a distributed cognitive framework for RAT selection for 5G HetNets. Here, the term RAT is used to describe a network attachment point (i.e., a Base Station, Access Point, or another terminal device) operating a certain wireless network technology. Each RAT is considered independently by our framework even when several of them are concentrated at a single physical node. Our proposed solution implements machine learning algorithms in order to satisfy the three user association perspectives above while providing a meaningful approach for multi-parametric modeling that echoes the principles of [10]. Our solution advocates the use of cognition at the device-level in order to learn optimal, or at least reasonably well-performing, decision policies based on the experience of the device itself.

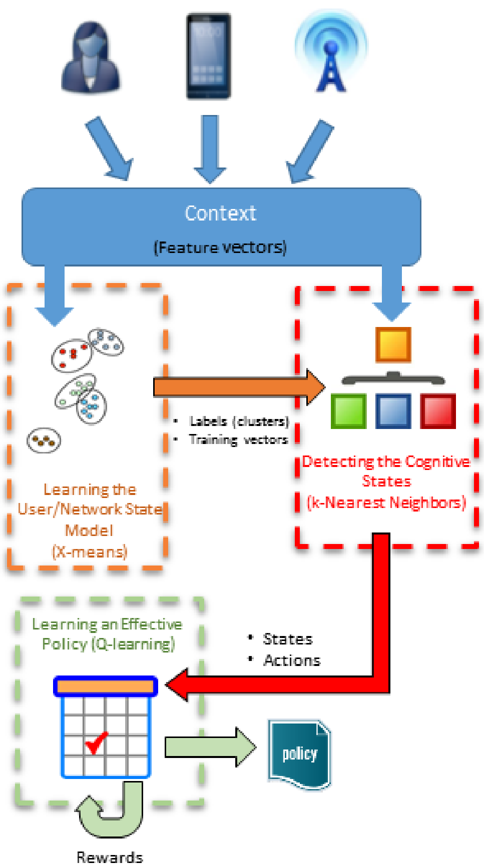


Fig. 1. High level overview of the proposed solution.

Figure 1 presents an overview of our proposed solution. As observed, it can be segmented into three stages: (1) Learning the user/network state model. (2) Detecting the cognitive states. (3) Learning an effective policy. Our final goal is to obtain a decision policy to pick the best action (i.e., to associate with a RAT) given the state of the terminal. For that, we propose to use reinforcement learning (specifically, Q-learning). The set of all possible system states $\mathcal{S}$ is the set of clusters from the first stage, obtained using the X-means algorithm (see Fig. 1). The current state s_t is the mapping of a current feature vector observed by the terminal device to one of those clusters. This mapping is done using the k-Nearest Neighbors (kNN) algorithm. We define the set of actions to be the set of available RATs.

The main contributions of this paper are: (1) A framework with intrinsic modularity for handling the user association problem. (2) The suggested use of reinforcement learning over supervised machine learning due to its low computational complexity and flexibility as a method for learning an effective user association policy. (3) The proposition for using unsupervised machine learning to formulate the user/network state model.

The novelty in the proposed approach is that the set of states are learned by the device itself and the number of states is also learned rather than pre-specified. This strategy can be really flexible in practice. This implies that a terminal, after collecting data during a learning period, may formulate a system characterization and optimize its own association decisions without any external intervention. In the following we will refer to these states as cognitive states, because they are supposed to be learned cognitively by each terminal device.

Sections II, III and IV of this paper discuss the conceptual foundations of each stage: Clustering of feature vectors for building a system state model, supervised classification to obtain the current system state, and reinforcement learning for learning good policies, respectively. Section V presents the results of simulations in which we have implemented the proposed algorithms for a simplified version of our solution, and section VI concludes the paper.

II. Learning the User/Network State Model

Formulating system models that are context-aware can be a daunting task, especially for our case of interest. The state definitions must be valid across the envisioned diversity of architectures, standards, applications and protocols, and deployment needs of future 5G networks; also, the state representations must be reasonable for reducing the overall processing and memory requirements.

It is desirable to build meaningful user/network states at the terminal level. As illustrated in Fig 1, in the first stage of our framework, we propose clustering of appropriately defined feature vectors as a method for autonomously constructing suitable system models for the user association task. The choice of the standardized fields for formulating the feature vectors is extremely important and requires considerable domain knowledge. This approach allows the system models to be custom-made for each node because the cognitive states learned will depend on the specific situations it has experienced. Note also that handling all this information at the device-level dramatically decreases the overhead requirements and supports the implementation of low-complexity algorithms that try to optimize the gains of individual user according to its particular needs.

We, however, acknowledge that some network state information that could be crucial for globally optimal user association may not easily be perceived by the terminal device (e.g., base station load and backhaul state). We propose minimal modifications of the standards for sending this information through broadcast channels transmissions.

A. The feature vectors.

The user state at any given time n is defined in terms of a collection of parameters that characterize the mobile device and RAT situation in the network at that particular time. Such parameters can be grouped as a tuple $\mathbf{x}$ of d descriptors or features. Thus, we may represent an observation of the user/network state as a point in a d-dimensional feature space: $\mathbf{x} = \{x_1, x_2, ..., x_d\}^T$, for $\mathbf{x} \epsilon \mathbb{R}^d$.

The mobile device needs to collect relevant statistics that reflect both network state and user needs/behavior information. The network state information could be obtained through Application Programming Interface calls to the terminal's network interfaces and broadcast messages, while estimating the user behavior might require sensing and packet-level analysis.

B. Clustering.

From a machine learning perspective, the feature vectors are simply patterns. These patterns can be divided into a set $\mathcal{S} = \{1, ..., M\}$ of M groups of similar characteristics or clusters, based on some measure of similarity. In our framework, these resulting clusters represent different cognitive states the terminal device can be in. Note that these states are derived from the past data observed by the terminal device. Thus, the proposed approach builds a system model relying on the multi-parametric context information of the feature vectors.

In this paper, we use the X-means algorithm to generate these cognitive states. X-means is a clustering algorithm that, in addition to classifying data into a set of clusters, attempts to estimate the number of clusters M from data itself. In our proposed solution, X-means runs off-line, utilizing the training feature vectors collected from all the different RATs. We have observed in our tests with X-means, that under certain conditions, the resulting state space $\mathcal{S}$ is clearly segmented into regions of clusters, each associated with a particular RAT. This reveals an effective characterization of distinct cognitive states. In other words, if we assume that there are only 3 RATs available, without loss of generality, we may formalize our definition of a cognitive state s as follows:

$$s \in \mathcal{S} = \{s_1^{(A)}, \cdots, s_{M_1}^{(A)}, s_1^{(B)}, \cdots, s_{M_2}^{(B)}, s_1^{(C)}, \cdots, s_{M_3}^{(C)}\},$$

where $M_1 + M_2 + M_3 = M$ is the total number of cognitive states and each element in the set $\mathcal{S}$ represents a cluster of feature vectors associated with one of the available RATs (in this case, A, B or C).

III. Detecting the Cognitive States

At time t, each device must determine the best RAT association based on an observed feature vector $\mathbf{x}_t$. Although there exist many supervised classification techniques, in our framework we use the kNN algorithm. The objective of this stage is classifying a new feature vector $\mathbf{x}_t$ into one of the cognitive states represented by clusters created by the X-means algorithm in the previous section. The kNN rule assigns the new feature vector $\mathbf{x}_t$ to the m-th class to which the majority of k closest training feature vector(s) belong.

As depicted in Fig. 1, in our proposed approach, the training vectors and classes are supplied by the clustering stage preceeding the kNN classifier. Let $\mathbf{y}_t$ be a tuple of feature vectors collected at decision time t by the terminal device:

$$\mathbf{y}_t := [\mathbf{x}_t^{(A)}, \mathbf{x}_t^{(B)}, \cdots]^T,$$

where $\mathbf{x}_t^{(\cdot)}$ denotes a feature vector collected at time t by the terminal device, associated to an available RAT.

The classification rule $f : \mathbf{x}_t^{(\cdot)} \to s^{(\cdot)} \in \mathcal{S}$, is a deterministic function, that maps each feature vector in $\mathbf{y}_t$ to a cognitive state (i.e., the clusters). Hence, the output of the classifier is essentially the set of possible cognitive states the device may be in at the next time instant.

IV. Machine Learning based User Association

A. States and actions.

We assume that the terminal device is able to recognize the cognitive state that corresponds to the currently active RAT association and denote it by $s_t \in \mathcal{S}$. Let $\mathcal{A}_t$ denote the set of actions available to the device at time t, where actions $a_t \in \mathcal{A}_t$ are identified as the available RATs.

B. Reinforcement learning based RAT association.

Reinforcement learning refers to a category of unsupervised machine learning techniques useful for learning an effective sequence of actions (i.e., a policy) to achieve a goal. In reinforcement learning, a decision-making agent receives a reward (i.e., a feedback) based on the action it chooses. In selecting the next actions, the agent tries to find a balance between exploration (choose untested actions) and exploitation (selection of actions already identified as beneficial) in order to reach at a globally optimal policy for maximizing the rewards.

As shown in Fig. 1 and mentioned in our introduction, the third stage of our proposed framework relies on the model generated in the first stage to learn a good association policy. In this paper, we propose the use of the Q-learning algorithm for this reinforcement learning stage. The Q-learning algorithm maintains a table of values that represent the goodness of taking a particular action when in a given state. Each table entry, $Q(s_t; a_t)$, is associated with a state-action pair, where $s_t \in \mathcal{S}$ and $a_t \in \mathcal{A}_t$. In our case, each Q-value is a measure of the "quality" of switching the currently active RAT association to either a different RAT or keeping it unchanged.

Since the user association problem requires multi-objective optimization that jointly maximizes the user perceived average throughput and QoS while minimizing service interruptions due to mobility conditions, we define our reward function as

$$R_t(s_{t-1}, s_t, a_{t-1}) = r_t \cdot U(r_t) \tag{1}$$

and

$$\begin{aligned} r_t = \;& \beta \cdot g(\text{Avg\_Measured\_Throughput}) - \lambda \cdot h(\text{HTTP\_RTP\_RTT}) \\ & - \zeta \cdot c(\text{Handovers\_Calldrops}), \end{aligned}$$

where $R_t(s_{t-1}, s_t, a_{t-1})$ is the delayed reward function computed at instant t that evaluates the consequences of the action a_{t-1} taken at instant $t-1$ while in state s_{t-1} that led to the current state s_t; $U(\cdot)$ is the Heaviside step function; β, λ and ζ are coefficients defined on the interval [0,1]; and $g(\cdot)$, $h(\cdot)$ and $c(\cdot)$ are suitably defined non-decreasing reward and cost functions of network performance metrics. $g(\cdot)$ and $c(\cdot)$ are restricted to be non-negative.

The Q-values are updated as:

$$\begin{aligned} Q(s_{t-1}, a_{t-1}) \leftarrow & (1-\alpha) Q(s_{t-1}, a_{t-1}) \\ & + \alpha [R_t(s_{t-1}, s_t, a_{t-1}) + \gamma \max_{a_t} Q(s_t, a_t)], \end{aligned}$$

where $Q(s_{t-1}, a_{t-1})$ denotes the Q-table entry defined by taking action a_{t-1} while being at state s_{t-1} at decision instant $t-1$; $\alpha : 0 < \alpha < 1$ is called the learning rate; and $\gamma : 0 \leq \gamma < 1$ is called the discount factor.

Once the Q-table is learned, the actions are selected as:

$$a_t^* = \begin{cases} \underset{a_t}{\text{argmax}}\; Q(s_t, a_t) & \text{, with probability } 1-\epsilon \\ P(\mathcal{A}_t) & \text{, with probability } \epsilon \end{cases},$$

where $\epsilon : 0 \leq \epsilon \leq 1$ is called the exploration rate; and $P(\mathcal{A}_t)$ is some probability distribution over the set of actions $\mathcal{A}_t$ defined using heuristics for exploration purposes.

V. Simulation

In this section, we present the results of MATLAB simulations that evaluate the proposed framework for user association in a multi-agent environment. The cognitive state models for each client node in our simulations were created using 3-dimensional feature vectors formed by the descriptors Peer ID (i.e., BSID index), downlink SINR, and the downlink Cell Load (i.e., average sampled radio resource utilization). An example is presented in Fig. 2, where each cluster represents regions of combinations of DL SINR and Cell Load across two different network attachment points.

Our simulation compares the network behavior of different user association mechanisms under common radio-frequency (RF) and node mobility conditions. We restricted our implementation to the DL transmission and have assumed that there is no interference. At each simulation time step, the SNRs of the client nodes relative to each serving node are recomputed according to (2):

$$SNR_{dB} = P_{dBm} - PL(d)_{dB} - N_{dBm}, \tag{2}$$

where $P_{dBm} = 40.0$ is the serving node transmit power in dBm, $PL(d)_{dB}$ is the path loss of the wireless link assuming log-normal fading ($n = 5.1, \sigma = 8.0$ dB) [11], and $N_{dBm} = -94.0$ denotes the client node Noise Floor in dBm.

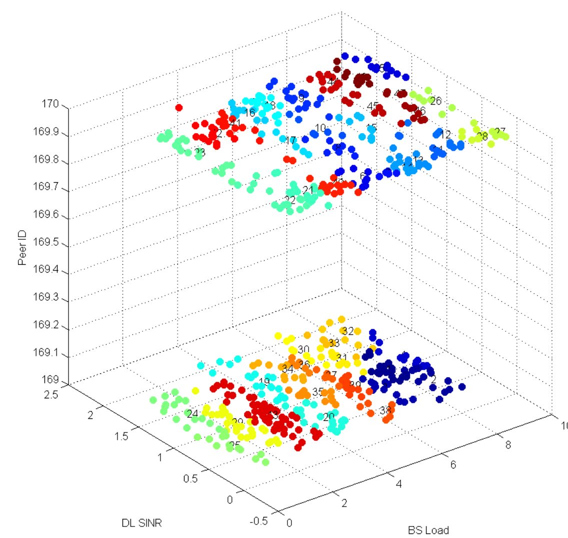


Fig. 2. X-means clustering of feature vectors formed by Peer ID, SINR and BS Load. Results in 48 groups that we may identify as 48 distinct cognitive states. 300 samples from Peer ID 169 and 300 samples from Peer ID 170 were used. SINR and BS Load values have been scaled by a factor of 0.1.

Let I and J denote the total number of client nodes and serving nodes in the simulation, respectively. Note that, the decisions of the i-th client node, for $i \in \{1, \cdots, I\}$ associated to the j-th serving node, for $j \in \{1, \cdots, J\}$, will affect its load. Let T_i^{req} be the required throughput of the i-th client node. Let T_{ij}^{max} be the mapping of SNR values to the maximum theoretical throughput according to the LTE standard [12]. Then, assume a time-sharing scheduling mechanism that allocates to each connected client node a time-slice according to (3). The average cell load L_j of the j-th cell is defined as (4).

$$w_{ij} = \frac{T_{ij}^{max}}{\sum_{i(j)} T_{ij}^{max}} \tag{3}$$

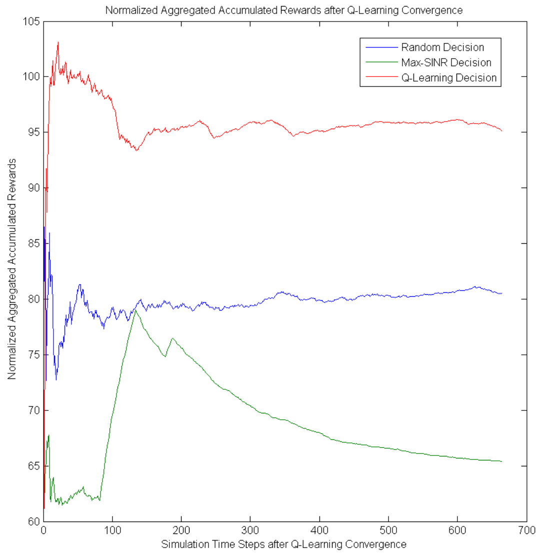
$$L_j = \sum_{i(j)} \frac{T_i^{req}}{T_{ij}^{max}} \tag{4}$$

In both (3) and (4), $\sum_{i(j)}$ denotes a summation over all the client nodes currently associated to the j-th cell.

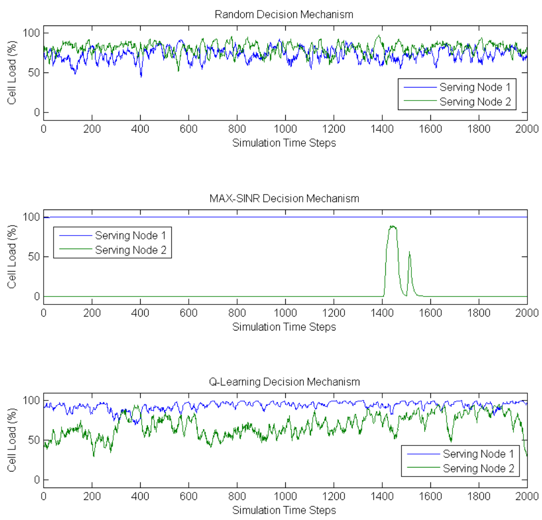
In our simulations, the individual rewards obtained by the client node i after each decision instant t, were computed using the following simplified form of (1):

$$R_t^i(s_{t-1}, s_t, a_{t-1}) = \beta \cdot g(T_i^{req}, L_j) \tag{5}$$

where $\beta = 0.001$ is chosen for convenience, and $g(T_i^{req}, L_j) = \begin{cases} T_i^{req}, & \text{if } L_j \leq 1.0 \\ T_i^{req}/L_j, & \text{if } L_j > 1.0 \end{cases}.$



(a) Normalized Network Rewards after convergence.



(b) Smoothed Cell Load per decision mechanism. $W = 21$.

Fig. 3. Simulation results for 3-client 2-serving node scenario. Q-Learning parameters: $\epsilon = 0.35$ before convergence and $\epsilon = 0.1$ after convergence, $\alpha = 0.3$, $\gamma = 0.7$, $\delta = 0.01$. Max-SINR parameters: $d_{thresh} = 10$ dB, $d_{hyst} = 5$ dB. kNN parameters: $k = 3$

Figure 3 presents results corresponding to one of our network simulations. We have considered 2 LTE serving nodes and 3 client nodes. The client nodes were configured using $T_i^{req} = 40.0$ Mbps, and a random walk-based mobility model was assumed. In Fig. 3, the aggregated rewards computed by Q-Learning are compared with the ones obtained by a random decision mechanism and with the max-SINR rule. Also, the smoothed cell load is presented for each case. The cell load presented was computed using (4), and values higher than 1.0 have been truncated. The smoothing operation, using a rectangular sliding window of size W, is necessary because the cell load can be very noisy and difficult to interpret when a large number of simulation time steps are used.

These results clearly show that the Q-learning outperforms both other decision mechanisms. Q-learning obtained, on average, $\sim$40% higher rewards than the max-SINR algorithm and $\sim$15% higher rewards than the random decision mechanism. This is, evidently, a direct consequence of achieving a better network load balancing. Moreover, on average, the random decision mechanism generated more than 3 times the overhead of the Q-learning mechanism.

VI. Conclusion

In this paper we have presented a distributed cognitive framework based on machine learning for the RAT association problem in 5G HetNets. Our proposal can learn simple state representations out of the terminal experience and user behavior, reducing the complexity of the core network design requirements. Also, it allows multi-objective optimization of the association decisions while incurring minimal network overhead. Our network simulation results showed the benefits of the proposed framework compared to alternative decision methods in a multi-agent environment.

Acknowledgment

This work was supported in part by the Air Force Research Laboratory, Space Vehicles Directorate, under grants FA9453-15-1-0314 and FA9453-16-1-0052.

References

[1] J. G. Andews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[2] B. Bangerter, S. Talwar, R. Arefi, and K. Stewart, "Networks and devices for the 5G Era," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 90–96, February 2014.

[3] H. ElSawy, H. Dahrouj, T. Y. Al-Naffouri, and M.-S. Alouini, "Virtualized cognitive network architecture for 5G cellular networks," *IEEE Commun. Mag.*, vol. 53, no. 7, pp. 78–85, July 2015.

[4] J. G. Andews, S. Singh, Q. Ye, X. Lin, and H. Dhillon, "An overview of load balancing in HetNets: Old myths and open problems," *IEEE Commun. Mag.*, vol. 21, no. 2, pp. 18–25, April 2014.

[5] V. Yazici, U. C. Kozat, and M. O. Sunay, "A new control plane for 5G Network architecture with a case study on unified handoff, mobility and routing management," *IEEE Commun. Mag.*, vol. 52, no. 11, pp. 76–85, November 2014.

[6] M. Zorzi, A. Zanella, A. Testolin, M. de Filippo de Grazia, and M. Zorzi, "Cognition-Based Networks: A new perspective on Network Optimization using Learning and Distributed Intelligence," *IEEE Access*, vol. 3, pp. 1512–1530, September 2015.

[7] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 414–454, First Quarter 2014.

[8] K. Shen and W. Yu, "Distributed pricing-based user association for Downlink Heterogeneous Cellular Networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1100–1113, June 2014.

[9] A. Y. Panah, S. Yeh, N. Himayat, and S. Talwar, "Utility-based radio link assignment in multi-radio heterogeneous networks," in *GC'12 Workshop: International Workshop on Emerging Technologies for LTE-Advanced and Beyond-4G, IEEE 2012*, Anaheim, CA, December 2012, pp. 618–623.

[10] F. Gao and K. Zhang, "Enhanced multi-parameter cognitive architecture for Future Wireless Communications," *IEEE Commun. Mag.*, vol. 53, no. 7, pp. 86–92, July 2015.

[11] B. Sklar, *The Mobile Communications Handbook*, 2nd ed. New York: CRC Press and IEEE Press, 1999, ch. 18.

[12] *3GPP TS 36.213*, Physical layer procedures (Release 9), Rev. V9.3.0, September 2010.