Use of Machine Learning for energy efficiency in present and future mobile networks

David Sesto-Castilla[*], Eduard Garcia-Villegas[*], George Lyberopoulos[+] and Eleni Theodoropoulou[+]

[*] Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

[+] COSMOTE Mobile Telecommunications S.A, Maroussi, Greece

Abstract— **Given the current evolution trends in mobile cellular networks, which is approaching us towards the future 5G paradigm, novel techniques for network management are in the agenda. Machine Learning techniques are useful for extracting knowledge out of raw data; knowledge that can be applied to improving the experience in the operation of such systems. This paper proposes the use of Machine Learning applied to energy efficiency, which is set to be one major challenge in future network deployments. By studying the cell-level traces collected in a real network, we can study traffic patterns and derive predictive models for different cell load metrics with the aid of different machine learning techniques. Such models are applied into a simulation environment designed to test different algorithms which, according to cell load predictions, dynamically switch on and off base stations with the aim of providing energy savings in a mobile cellular network.**

Keywords—5G; machine learning; energy efficiency

I. INTRODUCTION

Wireless communication is, by far, the fastest growing segment of digital communications, probably due to its cost-effectiveness and its ability to provide (near) ubiquitous access to a plethora of services and applications. Mobile cellular networks constitute one of the main pillars of the aforementioned growth: nowadays, the number of mobile subscribers exceeds the world population (7,600 million at the time of writing) and different studies predict a 5% annual growth rate (CAGR), reaching 9,000 million subscribers by 2021 [1]. Moreover, not only the number of subscribers is expected to keep growing, but also the amount of data traffic generated by each connected device is expected to show a ten-fold increase in the same period due to the proliferation of bandwidth-intensive applications (e.g. HD video streaming, virtual/augmented reality, etc.).

This scenario prompts the academia and the industry to work towards a new generation of wireless network, i.e. 5G. While the specification of 5G targets a time horizon beyond 2020, the requirements those new technologies are expected to meet have already been discussed and identified in different works [2], which foresee 10-100x device density, 10-100x typical user data rate and 5 times reduced latency. Those requirements are mainly tackled by new developments in radio access technologies (RAT), such as massive MIMO or mmWave. However, expectations of ~30% reduction in infrastructure energy consumption need a different approach. What is more, the aforementioned requirements should be met while keeping sustainable costs. This "implicit" requirement can be achieved through an innovative network architecture and intelligent management of the available resources. In this regard, techniques such as network function

virtualization (NFV), software-defined networking (SDN) and the application of big data-driven intelligence were also identified as key enablers of the future 5G [3].

In this context, the present paper explores the idea of applying machine-learning techniques to improve management procedures of future 5G networks towards a fully self-organized network (SON). Through the analysis of huge amounts of network-generated data, the proposed intelligent system is capable of anticipating the future state of the network so that appropriate actions can be taken in a timely manner. As a proof of concept, this paper studies the application of such intelligence to produce energy savings: we propose a mechanism to switch off/on cells according to their expected traffic conditions. These kind of techniques are highly relevant to mobile operators if we consider that densification (increasing number of deployed cells is needed to increase area capacity) will carry an increase in energy consumption and the fact that 70-80% of energy consumption (and gas emissions derived thereof) of mobile operators comes from their network infrastructure [4].

The remaining of this paper is structured as follows: section II motivates the use of machine learning applied to the management of mobile networks and reviews related literature. Next, section III discloses the proposed mechanism. Then, section IV describes the evaluation environment and V discusses the results. Finally, conclusions are provided in VI.

II. MACHINE LEARNING APPLIED TO MOBILE NETWORKS

A high level view of current mobile networks is given in Fig. 1, which shows three distinctive components: i) the user equipment (UE) is the device connecting to the access network and providing subscribers with access to their services; ii) radio access network (RAN) provides a wireless link to UEs through one or multiple cells served by multi-
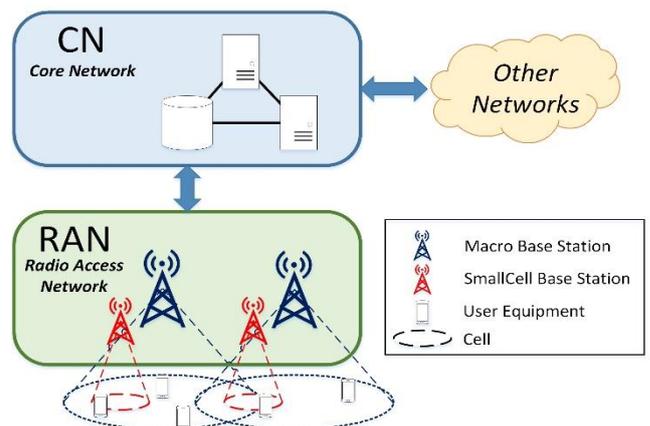

Fig. 1: generic architecture of a Mobile network

RAT base stations (BS) in order to reach the core of the network; and iii) the core network (CN) provides interconnection with other networks and is where final user services are managed.

In current deployments, especially in dense environments, the RAN provides a multi-layered heterogeneous cellular coverage, whereby small, low-powered cells are deployed within the area served by a macro-BS in order to increase the capacity in zones with higher traffic demands. With an even more intense "densification" expected for 5G networks, management of the RAN poses a great challenge, which will require a higher degree of automation. However, as suggested in [5], artificial intelligence (AI) could be useful to that aim. AI techniques, such as machine learning, are able to process large amounts of a variety of network-generated data (e.g. billing data, radio measurements, packet-level statistics, etc.) to produce exploitable knowledge with applications to network planning and optimization.

In [5], for example, authors suggest the use of cell-level traffic data to feed a classification system with potential applications to i) energy savings, by identifying low loaded cells as potential candidates to be switched off; and ii) spectrum planning, by identifying those cells that are likely to require additional spectrum resources. Similar applications could be derived from the work in [6], which evaluates regression mechanisms used to predict aggregated traffic from multiple cells. In [7], clustering of cells is used to improve traffic predictions. With the same objective, [8] combines clustering, wavelet decomposition and neural networks. Those works provide a good discussion on the potential uses of machine learning in 5G deployments; however, none of them defines or evaluates a specific mechanism.

III. A MACHINE LEARNING-BASED ENERGY SAVING SYSTEM FOR MOBILE NETWORKS

In [9], authors state that BSs (LTE's eNodeBs) can be either in an active state (handling UEs' data) or in an idle state (transmitting only downlink control signaling). However, the power consumed by the BS is almost the same in both states. Therefore, considerable energy savings could be obtained if the BS (or some BS's components) is switched off during idle periods. Applying this apparently simple concept is not trivial, though, since those idle periods are not deterministic but depend on the random distribution of traffic over time and space. As discussed in section II, however, machine-learning techniques could be used to predict the network behavior and identify those idle periods in advance.

In order to study the applicability of machine learning to the management of mobile networks, in this section we describe a proof-of-concept system intended to save energy by switching on and off a selected set of BSs depending on their predicted state in the next period of time. The operation of the proposed system is depicted in Fig. 2 and discussed in the following paragraphs.

A. Classification

Developing the idea suggested in [5], our approach is based on classification of cells. Classification is a supervised learning technique intended to associate an entity, which is defined by the evolution of its features (i.e. relevant metrics), with a given class, selected among a pre-defined set of classes. The classifier uses a training data set (i.e. a collection of entities and their features along with its proper classification) extracted from historical data to derive a model that will be used to identify classes from new incoming data. In this case, our entities are the BSs present in the RAN and the features considered are those metrics related to cell load, namely, the number of active users, carried traffic and physical radio blocks (PRB) assigned to UEs. For each new sample received from a given BS, the classifier determines its class for the next period of time. We defined the following three classes:

- *OFF:* BS candidate to be switched off the next period.

- *ACCEPT:* BS is not a candidate to be switched off and it is expected to carry a medium/low load.

- *FULL:* BS is expected to be highly loaded.

B. Clustering

The outcome of the classification is a model capable of predicting, with high accuracy, the state of a BS, according to its current load. In order to reduce the complexity of the system, instead of generating a classification model for each BS in the RAN, which does not scale when mobile network operators may control thousands of BSs, we use a reduced set of models. In that sense, we use clustering in order to identify sets of BSs showing similar behavior so that the same model (i.e. the same classification mechanism) can be applied to all of them without a significant loss of precision. Clustering is an unsupervised learning technique used to group entities into several clusters considering the similarity of those entities' features. The set of features considered by the clustering block of Fig. 2 consists of the average number of users, average throughput, average PRB utilization and their respective variances, obtained by processing historical data. Since BSs are likely to be switched off mostly during valley hours [9], we are interested in using specific models showing high precision for those periods. In consequence, we define clusters of BS according to their behavior in the period from 0 to 8am. We focused on those night hours since that period of time consistently showed a notable decrease in cellular traffic, according to our traces. However, note that whenever a BS experiences low traffic (i.e. similar to the behavior observed at night), it can be deemed as an *OFF* cell at any time of the day, not only nights.

That is to say, for each of the clusters identified, an independent classification model is generated (classifiers #1 to 3 in Fig. 2). Each classifier is then trained using data only from those BSs belonging to its corresponding cluster.
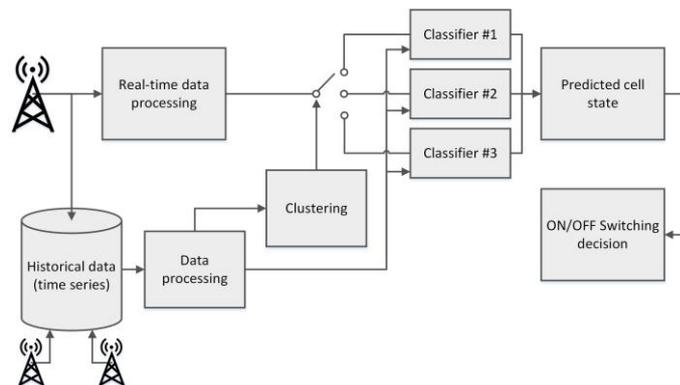

Fig. 2: design of a ML-based system for energy savings in mobile networks

Accordingly, the state of each BS is evaluated only by the classifier model corresponding to that BS's cluster. For example, if, according to historical data, a given BS is assigned to cluster #1, that BS' real-time data will be evaluated by classifier #1 to predict its state (OFF, ACCEPT or FULL) for the next period.

C. Switch off decision

Finally, the last block decides whether a given BS is switched off or kept in normal operation. This block receives periodic updates with information on the predicted state for each BS. Many kinds of algorithms, including other machine learning techniques, could be applied with the aim of maximizing energy savings while, at the same time, minimizing the impact over the final users. Note that a reduction in the number of active BSs may entail several issues, the selected energy-saving strategy should try to minimize: i) coverage gaps (i.e. areas not covered by BS) and, therefore, the possibility of leaving users without service; ii) reduced area capacity, possibly affecting QoS; and iii) an increased number of handovers because a cell could be serving UEs when the system decides to switch it off.

As a proof of concept, we propose a simple approach where the switch on/off decision takes into account the predicted state of a cell and its neighbors: if a cell is classified as *OFF* and a given number of neighboring BSs is in either *FULL* or *ACCEPT* state, then that cell can be switched off. In section V, we evaluate two versions of that algorithm:

- *Reckless algorithm*: switches off all BSs classified as *OFF*, regardless of the state of its neighbors. It is an aggressive approach used merely as a benchmark, which will provide larger power savings, but might affect the amount of users having to migrate or even left unserved if no other cell in the area is active.

- *Neighbor-aware algorithm (NA-X)*: before turning off a candidate cell, the state of neighboring cells in a given area is checked. If the ratio of active cells is higher than a given threshold *X*, the system is allowed to turn off that BS, hence reducing the probability of creating coverage gaps. Although we assume a fixed value, *X* could be set dynamically to deal with the trade-off between number of user migrations and the power saving goals.

IV. TOOLS AND DATA SETS

The study presented in this paper is based on simulation work, where the simulated scenarios are generated to recreate random variations of a real deployment in downtown Athens, Greece. Simulated BSs and UEs follow the behavior observed in the real deployment through a collection of cell-level traces.

A. Data sets

The data used in this study comes from the cell-level traces extracted from a real network, and counts a total of 10 LTE sites, each one having 1 to 6 sectors, providing a total of 27 cells. The combination of those cells in the aforementioned network serves an area of about 2km^2, each of them offering 20MHz of bandwidth and a coverage range between 100 and 600m. The data sets containing the traces from the network include 15 days of statistics with samples collected every 15 minutes, including weekends. The features that have been considered for the design of both the traffic

prediction models and the simulation environment are the following: date and time of the measured sample, cell ID, uplink (UL) and downlink (DL) data traffic measured in MB, average and maximum UL and DL cell throughput, average and maximum number of connected UEs, average Physical Resource Block (PR) usage per transmission interval, and modulation and coding scheme (MCS) distribution.

B. Tools

The studies presented in this paper required the use of different specific tools for each of the stages of the process. A brief summary of the main tools is given in the following:

RapidMiner [10]: an integrated data analytics platform that offers an environment for machine learning-related projects. RapidMiner Studio is a tool that offers a graphic interface in which, by means of boxes that perform different functionalities, data analytics projects can be programmed. RapidMiner has been used for the first stages devoted to analyze the data sets and the later extraction of prediction models for the different features studied in the project.

Simulation environment: with the objective of evaluating different management strategies in cellular networks, we developed a custom-made simulation environment capable of generating realistic scenarios from the traces extracted in a real network. The Java-based application developed for this project generates scenarios of 2,000 x 2,000 m, which are filled with the desired number of multi-sector BSs. BSs are distributed over the scenario in a semi-random way; that is, BSs are first distributed over a regular grid and then deviated randomly from that point and assigned a random coverage radius uniformly distributed between 100 and 600 m. Each cell is also assigned one out of three power profiles depending on its size [9] and assigned to a cluster with certain probability (following the same distribution found in the real network). In order to mimic the behavior of the real scenario, each simulated cell is bound to a real cell randomly chosen but belonging to the same cluster; in this way, each simulated BS will use data collected from a particular (real) BS, hence showing the same behavior of that real cell. Once the scenario is set, the simulation loops through the following steps to simulate a whole day for each BS in our scenario:

1. The features of each simulated cell are taken from the assigned real cell's traces; i.e. a given number of UEs is generated under the coverage area of each ON cell, where the carried traffic and the MCS distribution among those UEs follow that of its assigned real cell.

2. Upon a handover, the amount of PRBs needed by a UE that moves to another cell changes depending on the UE's carried traffic and new MCS distribution: we assume that a UE's carried traffic is maintained after a handover, however, its MCS may change in the new BS and, therefore, required number of PRBs will change accordingly (e.g. a worse signal quality from new BS will require a slower MCS, penalizing the utilization of the new BS with the use of more PRBs than in the old BS). To capture this effect, a penalty parameter *E* is computed for each cell according to equation (1), where PRB_t is the total PRB utilization in the cell in %, w_i is the weight of each of the *N* associated UEs, proportional to its MCS [1, 2 or 3] and t_i is the UE's throughput.

$$E = \frac{PRB_t}{\sum_i^N w_i \cdot t_i} \quad (1)$$

The PRBs required by a new UE arriving at a cell is then: $PRB_{new} = w_{new} \cdot t_{new} \cdot E$, where w_{new} depends on the MCS used by that UE after the handover and t_{new} is the offered throughput of that new UE in its old BS.

3. The cell state (*ON*, *FULL* or *ACCEPT*) for the next period (i.e. 15 minutes in this case) is predicted based on the data from the previous samples using the classification model (with a Random Forest approach) corresponding to each cell's cluster.

4. Switch on/off decision on each individual cell is taken according to the selected algorithm (either Reckless or NA-*X* version).

5. UEs (and their traffic) generated for a cell that has been switched off are handed over to the closest available cell in *ACCEPT* state, which is capable of satisfying the required PRBs (cf. step 2). If the UE is not within coverage of an *ACCEPT* cell, move to the closest *FULL* cell.

6. Compute relevant statistics: power consumption, number of migrated UEs, number of UEs left unserved, etc.

7. If the end of the day is not reached, load parameters from real traces corresponding to the new time of the day and go to step 1. Else, exit the simulation.

V. Evaluation and results

Using the tools described previously, in this section we evaluate the application of machine learning techniques (classification and clustering) within the context of the energy saving algorithm for mobile networks described in section III.

A. Classification

Classification and regression trees are known to perform automatic variable selection, which is an advantage when the set of features having more impact is not known. Besides, those models are capable of handling large data sets as this is the case in this work. Among decision trees, Random Forest has been found to be highly accurate [11].

Classification, as briefly explained in section III, is a supervised algorithm, which means that, in order to classify
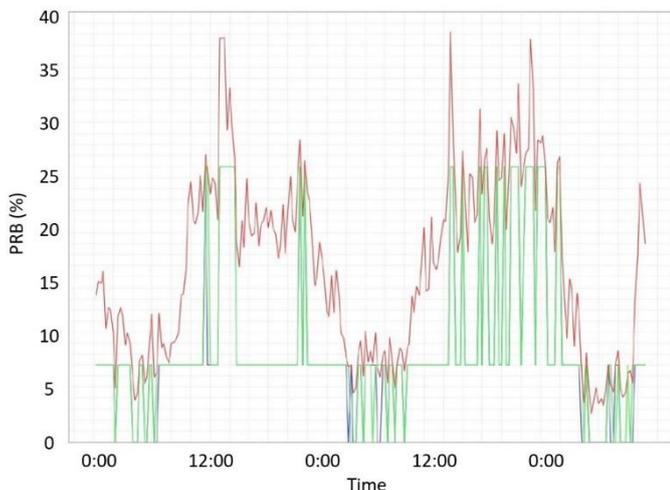

Fig. 3: Output of the classification (green) compared to the evolution in time of cell load (red).

entries by inspecting a given variable, it must have information about that same variable in the training set. The 'supervised' part of the mechanisms consisted in a priori processing of the historical data set in order to determine the future (i.e. in the next 15-minute period) state of a BS (OFF, ACCEPT or FULL) based on two thresholds $OFF < th1 < ACCEPT < th2 < FULL$.

Using the RapidMiner tool, we optimized the parameters of the Random Forest classifier: 10 trees, depth equal to 10 and window size 1. After different tests, we defined $th1 = 7\%$ and $th2 = 25\%$ of PRB usage, being those values representative of the load observed during valley hours. It is worth to mention that small variations of $th1$ and $th2$ around those values did not produce significant changes in the derived models. Fig. 3 shows a comparison between the classifier output (straight lines with three levels meaning OFF, ACCEPT and FULL from bottom to top) and simulated traffic. Table I illustrates the performance of the classification mechanism through a confusion matrix. It shows a high general accuracy of nearly 95%. The largest number of errors are found in OFF cells wrongly classified as ACCEPT (class recall of 77%), which, in fact, make the energy saving algorithm more conservative, as it reduces the chances of an active cell being turned off.

Table I: confusion matrix of classification algorithm

	true ACCEPT	true OFF	true FULL	Class precision
pred. ACCCEPT	68.4%	3.6%	0.0%	95.0%
pred. OFF	1.3%	12.1%	0.0%	90.3%
pred. FULL	0.4%	0.0%	14.2%	97.3%
class recall	97.6%	77.1%	100.0%	

B. Clustering

As explained earlier, clustering was used to identify groups of cells showing a similar behavior in order to reduce the number of models used in the classification process. For clustering, we use K-means since it is a simple method that works well for low dimensional data. In our case, the input of the clustering block (cf. Fig. 2) comprises the list of cells. In order to characterize each cell, we used historical data to obtain a measure of each cell's behavior. To that aim, we process the traces to obtain the average and variance of the PRB utilization, which seemed to be the most relevant metric according to the Random Forest classifier. Recall that, regarding the clustering process, we only considered values measured from 0 to 8am.

Being an unsupervised learning algorithm, K-means does not need any training, although it does need to know the desired number of resulting clusters (*k*). In this regard, different values of *k* were tested to reveal that there are two groups of cells highly differentiated from the rest: a group of cells carrying very high load and showing a highly variable behavior, and a group of cells carrying high/medium load also showing large variations. Increasing *k* (i.e. the number of clusters) did not change significantly the composition of those two groups and only created more clusters among the cells carrying a lower load and showing fewer variations.

Therefore, the value k = 3 was finally chosen. As a result, 7% of the cells were grouped as highly loaded cells, 26% as high/medium loaded cells and the remaining 67% as medium/low loaded.

C. Energy saving algorithm

Two strategies for energy saving (i.e. *reckless* and *NA-X*) are tested in the simulation environment described in IV.B. Both approaches are studied in two types of scenarios: i) *present scenario*, based on the deployment where real traces are extracted (cf. section IV.A); and ii) *future 5G deployment*, based on the same LTE deployment, but scaled up to represent a future 5G scenario, as envisioned in [12] (HetNet / Outdoor small cell scenario). The same LTE-based energy model from [9] is used in both cases.

- *Present scenario*: 90 cells placed randomly, with coverage radius assigned between 100 and 600 m and behaving according to the network traces, as explained in IV.B.

- *Future 5G deployment*: multi-tier deployment consisting of a first layer of 100 three-sector macro-cells (radius between 100 and 300 m) covering the whole area, and a lower layer of small cells (600 BSs with radius between 10 and 60 m) intended to improve area capacity. Note that macro-cells only consider other macro-cells when applying the *NA-X* while small cells consider both.

1) Present scenario

The main aim of the cell-switching strategies proposed in this paper is to reduce the power consumption in the network. However, turning cells off has a direct implication over the connectivity of the users, as those that are disconnected from a cell will have to migrate to another one, serving a partially overlapping area. Hence, there is always a trade-off between the power savings and the amount of migrated users –which has an impact over the signaling in the network and the user experience-. In the case of the first scenario, based on present deployments, the resulting density is 2.7 cells, that is to say,
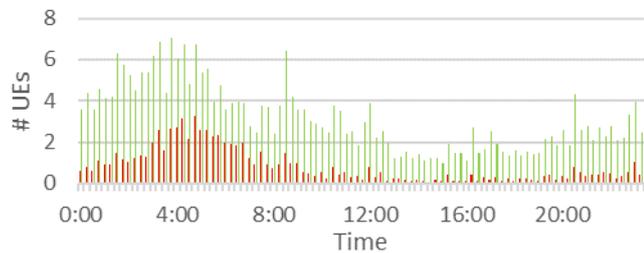
each single point of the scenario is served by 2.7 cells, on average; i.e. whenever a cell is switched off, UEs served by that cell will be able to choose between two possible new cells. This value is reduced to a different extent depending on the energy saving algorithm applied, causing coverage gaps in some cases and leaving some users without connectivity.

The graphs represented in Fig. 4 offer a visual comparison of the impact over user migration. On the left, the evolution over time of the migrated and unserved users with the *reckless* algorithm, showing an aggressive behavior; due to the large number of cells turned off, there are many users being forced to migrate to a neighboring cell, and some are left unserved due to coverage gaps (0.6%). As expected, the highest values of these two metrics are found during the night, time during which the load decreases and more cells can be switched off. On the other hand, the right graph shows the results of the *NA-70* algorithm, with lower values for both metrics, which are more equally distributed throughout the simulated day.

The behavior of the UEs detailed above corresponds to the ratio of on/off cells in Fig. 5. The *reckless* algorithm presents a high ratio of sleeping cells during night hours (more than 65%). A more conservative approach, such as the *NA-70*, shows a similar behavior in time, although clearly smoothed. Note that the energy consumption is directly proportional to the number of active cells. On average, the *reckless* version saves, near 40% of energy, and the *NA-70* around 27%.

2) Future 5G deployment

The second proposed scenario refers to a possible future deployment, with a higher cell and UE density. With a density of 3.3 cells (each point of the scenario is served by 3.3 cells, on average), the probability of causing coverage gaps is reduced and, therefore, the total figure of unserved users can be minimized if the algorithms are tuned properly.
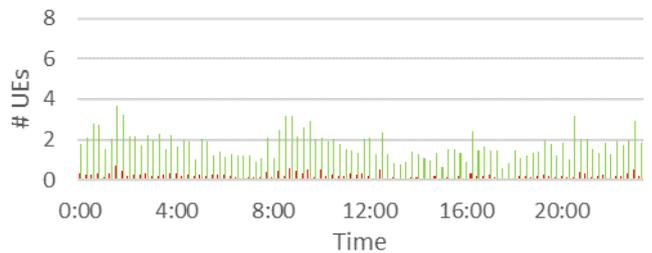

Fig. 4 Evolution over time of migrated (green) and unserved (red) UEs in the *present scenario* with the application of the *reckless* (left) and the *Neighbor-Aware-70%* algorithm (right).
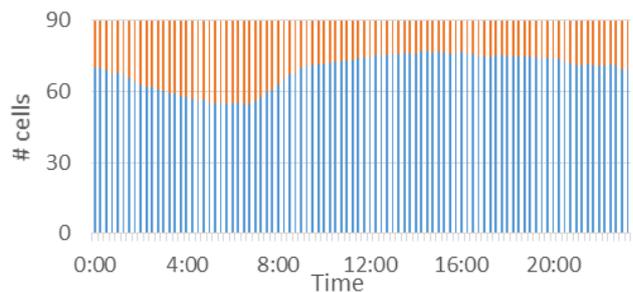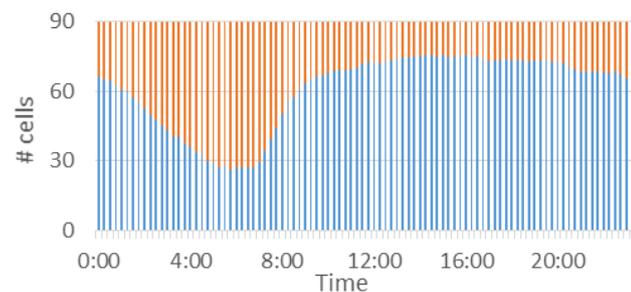

Fig. 5 Evolution over time of the on (blue) and off (orange) cells in a *present scenario* with an application of the *reckless* (left) and the *neighbor-aware-70%* algorithm (right).
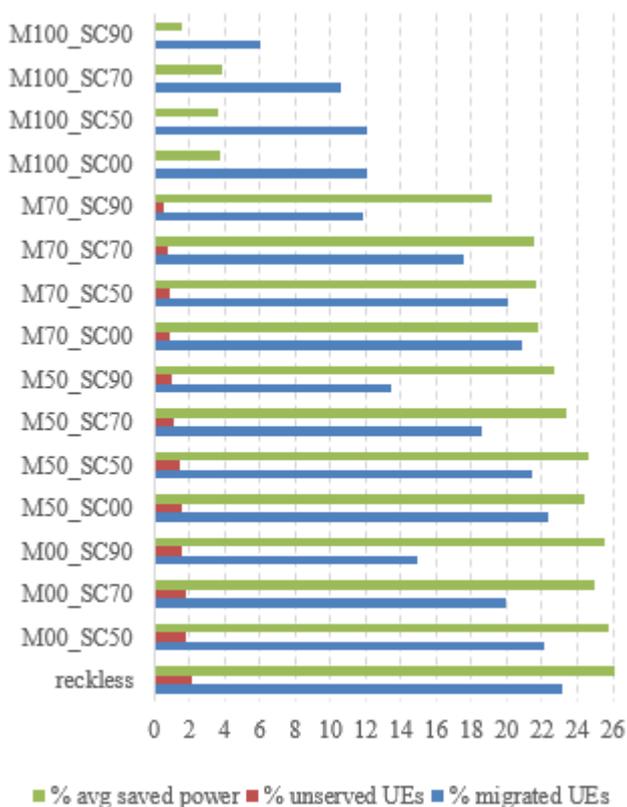
Fig. 6 performance figures for different *NA-X* algorithm
configurations in a future, 5G multi-layer deployment

In Fig. 6, we show a comparison of the main metrics obtained with different algorithm configurations, where the macro and small cell layers use different parameters for the *NA* algorithm. The nomenclature *MX_SCY* identifies the threshold parameters *X* and *Y* for both the macro (*M*) and small cells (*SC*), respectively. Note that a *NA-0* is equivalent to the *reckless* approach, while a *NA-100* means that no cell is turned off. The figure shows that more restrictive configurations -i.e. higher threshold values- entail less users being migrated and/or left unserved. In the case of M100_SCY (i.e. macro cells never sleep), the number of UEs unserved is always zero since 100% of the macro layer is always on service, leaving no coverage gaps. However, those configurations perform worse in terms of consumed energy (around 4%, on average, resulting in ~80.0 kWh saved per day). Also note that the particular case *M100_SC00* (macro BSs are never switched off and *reckless* small cells are switched off whenever they are in OFF state) is equivalent to the "variable network deployment" proposed in [13].

In the other extreme, when both the macro layer and the small cells run the *reckless* version of the algorithm (i.e. *M00_SC00*), energy savings exceed 26% (equivalent to ~500 kWh per day in the simulated scenarios). On the other hand, UEs left without service can exceed 1%. Note that small cells have a low impact on the overall power consumption and that important savings are witnessed only when the macro-layer participates in the energy saving strategy.

VI. CONCLUSIONS

This work discusses the application of Machine Learning techniques to improve network management in present and future mobile networks. Specifically, a proof of concept with the aim of improving energy efficiency is presented and evaluated. First, thanks to a collection of cell-level traces corresponding to a real LTE deployment, we identified ML techniques that could be useful in the context of mobile networks: clustering, used to identify cells with similar behavior, and classification, to predict the state of the network based on real-time measurements. Such models are then applied into a simulation environment designed with the purpose of testing different configurations of an energy saving algorithm, which switches cells off and on depending on the predicted network state. Finally, those algorithms are studied in two different types of scenarios, one representing present deployments, and a denser version, expected to be representative of upcoming 5G networks. According to the simulations, energy savings could translate into a cost reduction between $1,000 and 6,000 per year and per km^2 (current prices in Europe). Results also show that switching on/off small cells will not be enough to meet the required reduction in energy consumption. However, switching off at the macro layer may affect the service quality due to coverage gaps, which is an unacceptable outcome for a mobile network operator. Alternatively, macro-cells could switch off specific RATs (e.g. 2G, 3G, etc.) to reduce some energy and still be able to serve users. That approach, along with the characteristics of upcoming 5G BSs (i.e. gNB) will require a more sophisticated energy model and is left for a future work.

REFERENCES

[1] Ericsson, "Ericsson Mobility Report," 2016. [Online]. Available: https://www.ericsson.com/res/docs/2016/ericsson-mobility-report-2016.pdf

[2] M. Fallgren, B. Timus, et al., "Updated scenarios, requirements and KPIs for 5G mobile and wireless system with recommendations for future investigations", D1.5 of METIS project, March 2015.

[3] P. K. Agyapong et al., "Design considerations for a 5G network architecture," IEEE Commun. Mag. vol. 52, no. 11, pp. 65–75, 2014.

[4] GSMA, "Mobile's Green Manifesto", 2nd ed. June 2012.

[5] J. Pérez-Romero, J. Sánchez-González, O. Sallent, R. Agustí, "On Learning and Exploiting Time Domain Traffic Patterns in Cellular Radio Access Networks," in Machine Learning and Data Mining in Pattern Recognition. LN in Comp. Science, vol 9729. Springer, 2016.

[6] U. Paul, L. Ortiz, S. R. Das, G. Fusco, M. Madhav Buddhikot, "Learning Probabilistic Models of Cellular Network Traffic with Applications to Resource Management", DYSPAN 2014.

[7] H, Maciejewski, M, Sztukowski, B. Chowanski, "Traffic Profiling in Mobile Networks Using Machine Learning Techniques", Digital Information Processing and Communications. Communications in Computer and Information Science, vol 188. Springer, 2011.

[8] Y. Zang, F. Ni, Z. Feng, S. Cui, Z. Ding, "Wavelet Transform Processing for Cellular Traffic Prediction in Machine Learning Networks", ChinaSIP 2015.

[9] K. Hiltunen, "Utilizing eNodeB Sleep Mode to Improve the Energy-Efficiency of Dense LTE Networks", in Proc. of 24th IEEE Personal Indoor and Mobile Radio Communications (PIMRC), 2013.

[10] RapidMiner official website - https://rapidminer.com

[11] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?, in Journal of Machine Learning Research, 15(1), pp. 3133-3181, 2014.

[12] M. Maternia, J.F. Montserrat, et. al. "Performance evaluation framework," D2.1 of METIS-II project, January 2016.

[13] K. Hiltunen, "Improving the energy-efficiency of dense LTE networks by adaptive activation of cells," in Proc. of *IEEE International Conference on Communications (ICC)*, pp. 1150-1154, 2013.