A Machine Learning Mechanism for Improved Energy Consumption in 5G Small Cells

Allayiotis Foivos, Bouras Christos, Kokkinos Vasileios
Computer Engineering and Informatics Department
University of Patras
Patras, Greece
st1056636@ceid.upatras.gr, bouras@upatras.gr, kokkinos@cti.gr

***Abstract*— By introducing 5G networks in our lives, we are faced with the challenge to lower energy consumption, which is expected to increase significantly. As a result, the communications sector must devise solutions to improve the energy efficiency of 5G networks while maintaining spectrum performance. We propose a Machine Learning Mechanism that provides new sleep mode functionalities, that reduce energy consumption during idle hours, on small cells. We will describe our model and examine its performance using data collected from a simulated environment. Furthermore, we present the results of our findings and conclude that such models are necessary for the introduction of 5G networks. Lastly, possible future steps, are discussed.**

Keywords—5G, Machine Learning, Energy Efficiency, Small Cells

I. INTRODUCTION

The exponential increase in data traffic has resulted in the expansion of mobile networks. This increase, led to radio network energy consumption becoming a substantial contributor to operator electricity use and operating costs. In a study in Finland, mobile networks were responsible for 0.7% of the annual electricity consumption [1]. Past studies show that base stations are responsible for 57% of power consumption [2]. It is in the hands of the operators to distribute solutions that minimize the energy consumption of Base Stations (BSs) and small cells. According to [3], 5G subscriptions in 2020 were forecasted at 220 million, while by the end of 2026 they predict 3.5 billion 5G subscriptions, making around 40% of all mobile subscriptions. We believe that using a Machine Learning (ML) Algorithm to predict the category of small cells, specifically femtocells, and make the decision whether to set cells to sleep, improves the energy efficiency. There are no remedies that we are aware of that combat this problem, using a technique like ours.

According to the author of [4], ML is an artificial intelligence (AI) technology that allows computers to enhance and develop from information without being particularly programmed to do so. Because 5G networks are becoming increasingly sophisticated, it has become a common practice to use ML Algorithms to carry on certain tasks, which if done manually, would be considered impossible. The authors claim in [5] that AI and ML in general will be an integral part of future wireless networks. The applications range from resource allocation and management to policy control, and security monitoring. Better processing and computing capacity, access to a vast quantity of data, and improved software methodologies all contribute to the development of AI-powered wireless connections, allowing for an intelligent Radio Access Network (RAN) and the proliferation of enormous AI devices. Incorporating AI capabilities into future networks will facilitate independent and self-functioning networks, that are able to adapt to changing circumstances in real time. We also seek to provide our simulated network with self-organizing capabilities. Our model will classify, each femtocell into one of three categories based on various characteristics. The model will then check the states of neighboring cells to decide what actions should be taken for that cell.

The work in [6] shows an example of ML algorithms used in 5G networks, where a branch-and-bound procedure (BB) is used to generate a large dataset for training an artificial neural network (ANN) to determine the best mapping between the network channel recognition and the corresponding optimal power control policy. By focusing on small cells, we develop a simpler, less complex model that can be used in real-time. In [7], the authors proposed a similar mechanism for Heterogeneous Cloud Radio Access Networks (H-CRAN), that provides an increase in energy and spectral efficiency. However, resource allocation in H-CRAN, considering the heterogeneity of the network, has a heavy toll on energy consumption while on small cell networks, energy consumption is lowered even further. On the contrary to [8], where an optimization algorithm is proposed, we believe that a ML model, will provide better, more accurate results, and in less time, as ML models require less time to provide results. Moreover, in [9] a cell zooming technique is proposed that proves to reduce energy consumption. Cell zooming reduces the capacity range of cells, by altering their capabilities. Even though, the authors manage to increase energy efficiency, altering the physical capabilities of cells increases the complexity of any suggested solution. A more elegant solution of switching off cells ensures that our mechanism is easily applicable and more efficient.

Different approaches can also be found in [10] and [11], where instead of advancing in energy-saving techniques, the authors chose to minimize the energy consumption required. Such studies are more recent but are a less powerful approach than mechanisms like ours. Even though the two approaches might appear similar, they generally lead to different allocations of resources. Our contribution is mainly focused on the classification of the cells, with the use of a Machine Learning Algorithm, and after that, we produce an additional algorithm that decides the actions that need to be made, to achieve the least energy consumption while maintaining the Quality of Service (QoS).

The remainder of the paper is laid out as follows. In Section II, we describe the system model we used. The proposed mechanism is described in detail in Section III and an example is also provided. Furthermore, in Section IV, we analyze our proposal after using it in a simulated environment and producing tables and graphs, with the results. Finally, the conclusions and future steps are provided in Section V.

978-1-6654-9792-3/22/$31.00 ©2022 IEEE

II. System Model

This study is based on the sleep mode model in [12]. The components that have been turned off are those that are not required for the back-haul network connection or in order to wake up the femtocell by sensing user activity. A sensor called "sniffer" monitors user activity by detecting increases in uplink received power, which indicates a user connection. If the sniffer's threshold is surpassed, the femtocell is woken, and a handover occurs. Both the necessity for an underlying macrocell infrastructure as well as a mobility management system are drawbacks. On the other hand, urban situations have little effect on the former. According to [13], the additional signaling caused by sleep mode integration and handover is more than reimbursed by the reduction in the activities of the femtocell, in sleep mode. This method can be used to turn off sections of the memory associated with the field-programmable gate array (FPGA) and the radio frequency (RF) transmitter. The power consumption of the "sniffer" is predicted to be 0.3W. The following sources can be used to create power savings:

$$P_{savings} = P_{micro} + P_{FPGA} + P_{receiver} + P_{transmitter} + P_{amplifier} - P_{sniffer} = 4.2W \quad (1)$$

In Long Term Evolution Advanced (LTE-A) investigated in [12], the core network identifies connections as well as sending a wake-up signal to nearby small cells that the user has access to via the mobility management entity, a detrimental part of LTE-A. Thus, it provides the advantage of powering off practically every element of the small cell except the CPU and backhaul circuitry, leading to a 70 percent reduction in power usage. In terms of performance metrics, we follow LTE-A recommendations, employing Orthogonal Frequency-Division Multiple Access (OFDMA) for flexible spectrum allocation and LTE-A guidelines for variables like route loss in urban contexts [14]. We consider the location of the femtocell while determining its power. Due to interference, the effective range of a small cell in co-channel tiers is determined by the distance between a macro and a femtocell. As a result, we modify the power of femtocells to maintain a consistent coverage radius [15]:

$$P_f = min(P_m + G - PL_m(d) + PL_f(r),\ P_{max}) \quad (2)$$

where P_m is the power of the macrocell nearest, G is the antenna gain and $PL_m(d)$ is the macrocell path loss at the femtocell distance d. Finally, $PL_f(r)$ is the path loss at the target radius r and P_{max} the maximum power. Equations (1) and (2) are used to calculate the amount of energy that will be saved.

The Signal-to-interference-plus-noise ratio ($SINR$) of user u on subcarrier k, is then based on the following equation:

$$SNIR_{u,k} = \frac{P_{B,k}G_{u,B,k}}{N_0\Delta f + \sum_{B'} P_{B',k}G_{u,B',k}} \quad (3)$$

where $P_{B,k}$ is the power on which the user's serving base station B on subcarrier k transmits, and $G_{u,B,k}$ is the channel gain between serving cell B on sub-carrier k and user u that the cell provides service to. $P_{B',k}$ and $G_{u,B',k}$ indicate the combined strength of all other interfering BSs (femtocell or macrocell) as well as the gain between them and the user u, respectively. N_0 denotes the white noise power spectral density, and Δf the sub-carrier spacing.

To calculate the user's capacity on that subcarrier [16] we then use the following:

$$C_{u,k} = \Delta f \cdot \log_2(1 + \alpha SINR_{u,k}) \quad (4)$$

where α is defined by $\alpha = -1.5/\ln(5BER)$. Setting $\beta_{u,k} = 0$ when the sub-carrier k is not allocated to user u $\beta_{u,k} = 1$ and otherwise. Furthermore, in order to estimate the system's throughput of the serving BS based on the subcarrier assignment, according to [17], the following is used:

$$T_B = \sum_u \sum_k \beta_{u,k} C_{u,k} \quad (5)$$

Another metric that we consider is QoS. The service we wish its quality to be maintained is the constant and uninterrupted connection of users to the network, as we put femtocells in SLEEP MODE. Therefore, the measure of comparison that we are interested in is, how many users, out of the total in the system, remain connected, after our mechanism is applied. With the above we can formulate an equation for QoS as follows:

$$QoS = \frac{\sum_{i=0}^{N} c_i}{N}, C_i = \begin{cases} 1, if\ connected \\ 0,\ otherwise \end{cases} \quad (6)$$

where C is an array that is filled with ones and zeros and N the total number of users. Each position in the table corresponds to a user id. If the user is connected to the network, then there is the number 1 in the position corresponding to the user id, while, if he is not connected, there is the number 0. Equations (3) – (6) are used to evaluate the user's experience.

Lastly, we should discuss the ML model that we will use. We will use a widely popular algorithm, known as Logistic Regression (LR). In both [18] and [19] LR is pitted against various other algorithms. The authors of [18], review the LR model, and how it performs when compared to other lesser-known algorithms. They concluded that in all cases, LR outperforms the rest, some by a wide, others by a small margin. Moreover, in [19] LR is reviewed alongside another established ML model, known as Support Vector Machine (SVM). Even though, SVM proves to be an adequate replacement, in some cases, LR still provides the better results in most cases.

A sigmoid function is used to forecast the outcome of the LR, which produces a number between 0 and 1. By setting various thresholds, we can predict the class a cell belongs to, based on the output of the sigmoid function. The classes are explained in the next chapter; but, for now, we map the classes as, 0 for SLEEP MODE, 1 for ACTIVE and 2 for FULL LOAD. We assume the output is: 0 when the sigmoid function produces an output less than 0.33, 1 if it produces a value ranging from 0.33 to 0.67, and 2 if it returns a value greater than 0.67. The sigmoid function is defined as such:

$$h = \frac{1}{1 + e^{-z}} \quad (7)$$

where z corresponds to the input characteristics, which are multiplied by θ, a randomly initialized value:

$$z = \theta X \quad (8)$$

The input feature is X in this case. When, many input parameters exist, this formula becomes as such:

$$X = \theta_0 + \theta_1 X_1 + \theta_2 X_2 + \theta_3 X_3 + \cdots + \theta_n X_n \quad (9)$$

The individual input parameters are certain measurements that can be extracted from the cells. Those are, the number of subscribers of a cell and the number of unique subscribers, meaning users that can be serviced from that specific cell. More measurements include the carried traffic that the cell exhibits, paired with the total throughput that it provides, as well as the overall energy that it consumes. When so many parameters are involved, the use of ML is apparent. Thus, trying to isolate and examine all the cases that might occur, would require massive amounts of memory and computational time. Apparently, the final pair of equations, (7) – (9), are used to design our model, that will carry out the predictions.

III. PROPOSED MECHANISM DESCRIPTION

Classification is a supervised learning technique. It is responsible for assigning a category to an entity, based on a certain set of features. In our case, an entity constitutes a cell. In this process, we will build a model that will take as input the characteristics of cells, to categorize them into three possible states. The metrics, that will be considered will be the number of users, carried traffic, total throughput per cell, and the overall energy consumed by a cell. The three possible states are:

- *SLEEP MODE:* A single cell can be turned off.
- *ACTIVE:* The cell can't be turned off; however, it should have a low to medium load.
- *FULL LOAD:* A full load is expected in the cell.

A cell can be categorized as SLEEP MODE, when it does not host many users, and most importantly when it does not have many unique users. Moreover, when the cell's traffic and throughput is relatively low. Next, a cell is classified as ACTIVE, when the number of users and unique users is at the middle of the cell's capacity. Also, when the traffic and throughput is relatively high, and when the energy is at an acceptable level. Lastly, the cell is considered FULL LOAD, when it is nearing the limit of its capacity, or when it cannot serve any more users. The amount of traffic and total throughput is also expected to be very high. The total energy that the cell exerts, is at its highest levels. For each new cell received, the Classification process determines to which category the cell falls upon.

Finally, the output of the Classification process is given as input to the Decision-Making process. It decides, based on the category of a certain cell, whether it should be put into sleep mode or carry on with its normal operation. The techniques that could be used in the Decision-Making process vary based on what results we wish to achieve. The algorithms we could employ for the Decision-Making process, could be one of the following:

GREEDY ALGORITHM: Such algorithms, choose to switch off each cell that was classified as SLEEP MODE. It does not consider the states of neighboring cells. In most cases, it achieves high amounts of energy saved, but it might leave users unserved. An example is shown below, where the classified network, will be classified as shown in Figure 1:

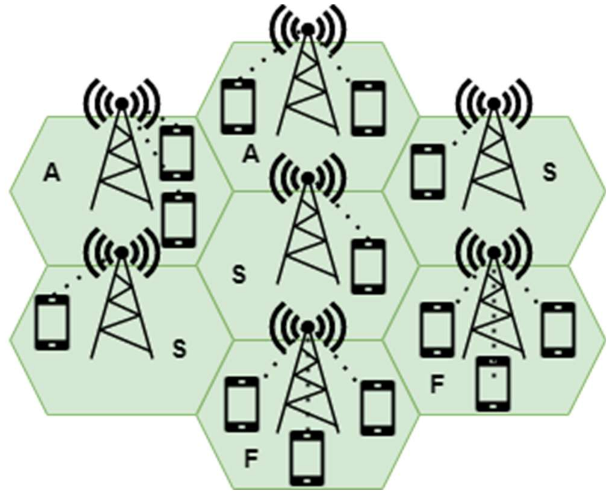

Fig. 1. The classified network

Where the letter S, A and F means that the femtocell was classified as being SLEEP MODE, ACTIVE and FULL LOAD accordingly.

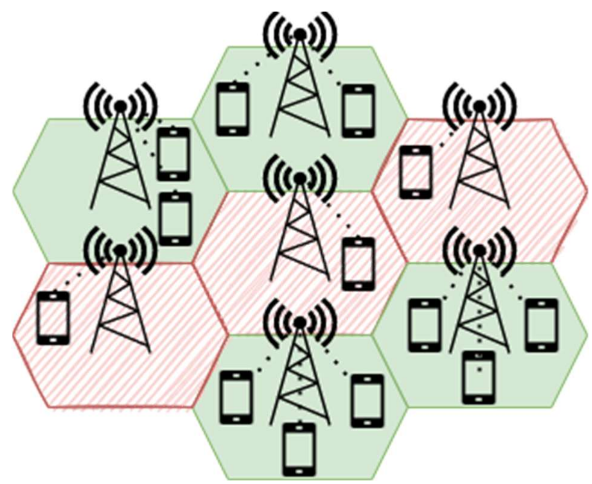

Fig. 2. The output network after the Decision-Making process using a greedy algorithm.

After the classification process is over, we receive as output, a matrix that holds the classification result of each cell, which we will consider as the classified network. The classified network is then given as input to the Decision-Making process. By using a greedy algorithm, all the cells that were classified as SLEEP MODE will be turned off and the network will have the form of Figure 2, where the femtocells that are colored red and sketched, were turned off. As we can see the energy spent by the network has been lowered by three femtocells, but several users will be left unserved.

- *NEIGHBOR-AWARENESS ALGORITHM:* Such an algorithm considers the states of the neighboring cells. If a neighboring cell is classified as ACTIVE, then the users are migrated from a SLEEP MODE cell to an ACTIVE CELL. After the Decision-Making process, the network will have the following arrangement:

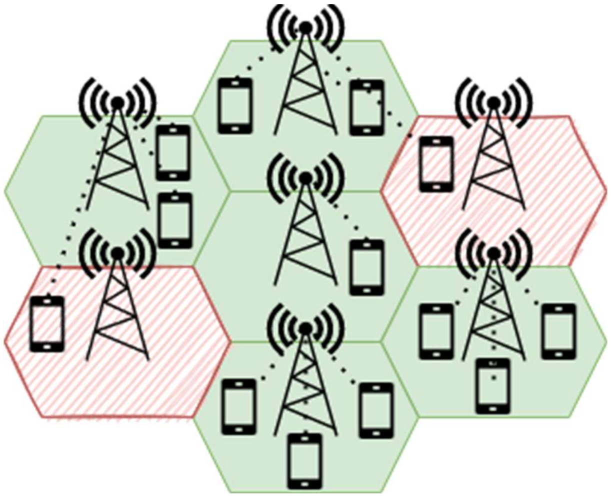

Fig. 3. The output network after the Decision-Making process using a neighbor awareness algorithm.

As we can see, two users were migrated to two neighboring cells. The total energy saved, might be less, but still very optimal as the total energy required is reduced by two femtocells, but no user is left unserved.

A simple approach would be used to turn off each cell that is categorized as SLEEP MODE. By using such aggressive techniques, we might see a large amount of energy saved, but we might leave a small number of users unserved, thus lowering QoS. Another safer approach would be an algorithm that checks the state of neighboring cells, of the cells that belong in the SLEEP MODE category. If the neighboring cells belong to the FULL LOAD category, the users cannot be migrated and the cells that were categorized as SLEEP MODE, cannot be turned off. If, on the other hand, the neighboring cells belong to the ACTIVE category, the users of cells' that were classified as SLEEP MODE are migrated to the neighboring cells and then switched off safely. Such approaches provide a sufficient level of energy saving, but also leave no user unserved. The mechanism can be summarized as a pseudocode like the algorithm below:

Algorithm 1 Energy efficient mechanism:

```
Collect data from the small cell network
Feed data as input to model
Retrieve classified cells as output
For each classified cell
    Check class of cell
    If cell_class == "SLEEP MODE"
        While #cells_checked < neighbouring_cells_size
            If neighboring_cell_state == "ACTIVE"
                Transfer users to neighbor cell
```

IV. Performance Evaluation

Our simulated network has nine macrocells, each with a base station broadcasting at 46dBm at the cell's center. Several femtocells, as well as their subscribers, are installed. Each femtocell may support up to fifty users at the same time. The transmit power of femtocells is estimated using the system model, with a maximum value of 20Bbm allowed. Several macrocell users were also dispersed across the region at random. Some of the parameters for simulation are listed in Table 1. The network parameters were gathered from the LTE-A 3GPP guidelines and the LTE simulator in [20].

We firstly considered an area within a radius of one kilometer. In this area multiple femtocells are employed, ranging from 2500 to 4500, which are used to provide coverage to a number of users, ranging from 5000 to 25000. The users are randomly deployed inside our simulated network, and in regard to their position, they are following either a normal distribution or a uniform distribution. Each femtocell can transmit up to 20 dBm and serve from 10 to 50 simultaneous users. Each macrocell site has numerous sectors and consumes 46 dBm of total power. For macrocells, no dynamic power saving options are expected. On macrocells, power consumption is constant regardless of supported traffic, however on femtocells, data traffic is critical.

TABLE I. SIMULATION PARAMETERS

Simulation parameters and Values	
Simulation parameters	***Values***
Macrocells	9
Macrocell radius	250 m
Femtocells	[2500, 3000, 3500, 4000, 4500]
Femtocell radius	10 m
Femtocell subscribers	10 - 50 (per femtocell)
Bandwidth	22 MHz
Carrier frequency	2.4 GHz
BS transmit power	46 dBm
FBS max transmit power	20 dBm
Total Users	[5000, 10000, 15000, 20000, 25000]

Each channel has its own carrier frequency. In our case, we have a carrier frequency of 2.4 GHz, which corresponds to 2437 MHz. Due to the fact that the bandwidth is 22 MHz, then the modulated signal on the channel will be in range from (2437 MHz - 11 MHz) to (2437 MHz + 11 MHz), i.e., from 2426 MHz to 2448 MHz.

TABLE II. CONFUSION MATRIX OF THE CLASSIFICATION MODEL

		Predicted		
		SLEEP MODE	ACTIVE	FULL LOAD
Actual	SLEEP MODE	53.33%	2.22%	0.00%
	ACTIVE	0.00%	20.00%	6.67%
	FULL LOAD	0.00%	0.00%	17.78%
	Precision	95.8%	74.9%	100.00%

In terms of evaluation, we must first look at our mechanism in detail, regarding its accuracy, areas where some of the accuracy is lost, and how much it impairs the performance of the overall mechanism. Starting with the accuracy, the output, which is the classification of femtocells, was split into a training and a testing set of samples, each accounting for 55% and 45% respectively. After the training was complete, our model has shown a prediction accuracy of the training set of 91%. Furthermore, using a confusion matrix, Table 2 illustrates a more detailed performance evaluation, of the classification component, of our overall mechanism.

In detail, out of the 45% of the testing set, 55.55% were originally SLEEP MODE femtocells, 26.67% were originally ACTIVE and 17.78% were FULL LOAD femtocells. Moreover, 95,8% out of the 55.55% were correctly predicted as SLEEP MODE. The 4.2% error rate occurs when SLEEP MODE femtocells where falsely classified as ACTIVE. Out of the percentage of femtocells, classified as ACTIVE, the 74.9% of them were correctly classified, and the remaining were falsely classified as FULL LOAD. Finally, all the femtocells that were classified as FULL LOAD were predicted correctly. The bulk of mistakes occur when an ACTIVE cell is incorrectly labeled as FULL LOAD. The reason behind, most of the errors occurring in those specific instances, is the fact that the cases are very similar to one another. A solution for this problem, could be adjusting the weights, of the algorithm, thus emphasizing more on certain metrics, rather than others. For example, increasing the weight of the number of unique users, such that the specific metric is considered more important, and should be accounted for the most. Nevertheless, this is a reassuring result since it makes the energy-saving strategy more cautious by lowering the likelihood of an ACTIVE cell being switched off.

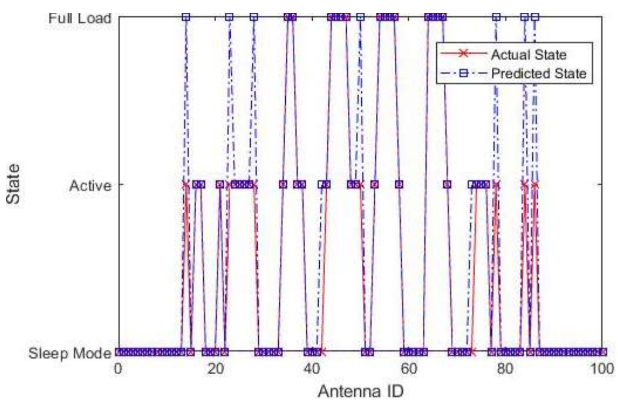

Fig. 4. Progression of the model, the actual versus the predicted states.

In Figure 4, we illustrate the comparison between the output of our classifier model (red line with circle marker), which is the predicted state of an antenna, and the true state of the corresponding antenna (blue line with diamond marker). In the Y axis, we can see the three possible states, that were discussed in Section III, which are SLEEP MODE, ACTIVE and FULL LOAD. The points at which the two graphs are tangent are the points where the output of the model, which as we have said is the predicted state of an antenna, or the BS inside a cell, corresponds to the actual state of that antenna. The cases in which the graphs are not the same are the cases where the model made a mistake and did not correctly predict the condition of the antenna. These cases as we can see are minimal in front of the overall sample. The following graph can be seen as an indication of the accuracy of our model.

Figure 5 represents the progression of the amount of energy saved, as the number of users increases inside the simulated network. The energy saved is measured as a percentage of the total power consumed in the network. What we can most clearly see, is that in all cases the Greedy version, achieves a higher percentage of energy savings, in contrast to the Neighbor Awareness algorithm. More specifically, the Greedy Algorithm, can save massive amounts of energy, reaching nearly 80% in certain conditions. On the other hand, the Neighbor Awareness Algorithm peaks at 68% of the total power, saved, in perfect conditions. Furthermore, it can also fail to save any energy at all, in conditions that are consider far from ideal. What stands out even more, is the fact that the ideal conditions for the Greedy Algorithm and the Neighbor Awareness Algorithm, differ greatly.

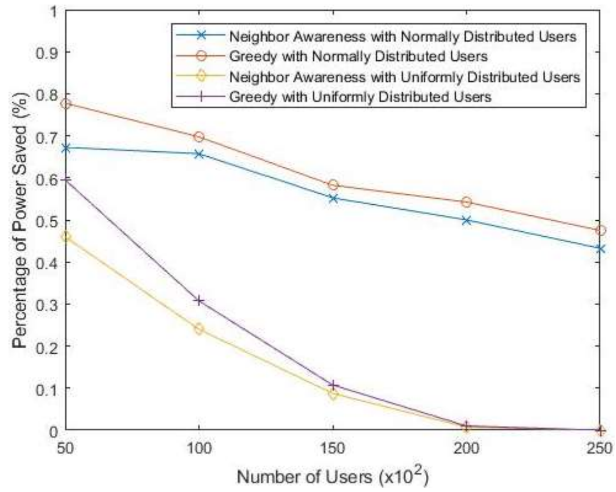

Fig. 5. Percentage of Energy Saved per Number of Users.

Regarding QoS, Figure 6 illustrates the advancement of the QoS, and the second variable is yet again the Number of Users. The QoS is calculated as shown in the Chapter II, thus it is a percentage that represents the proportion of users that are being serviced, over the overall number of users, in the cases we examine.

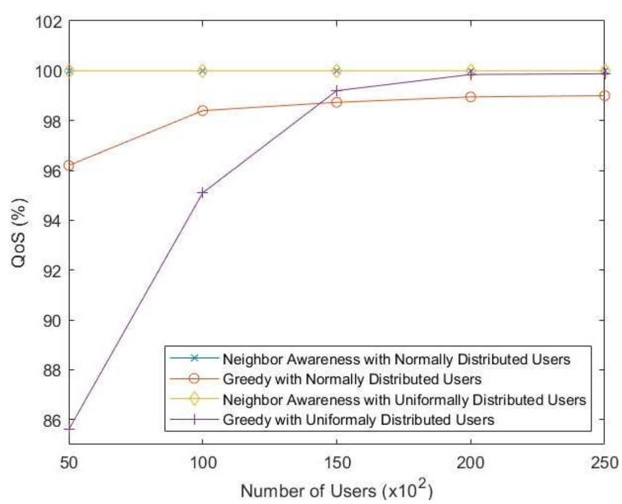

Fig. 6. QoS per Number of Users.

In this figure, we can distinguish that the Neighbor Awareness Algorithm, vastly outperforms the Greedy version of the final component of our mechanism. On all conditions, the QoS is maintained at 100%, thus not even a single user loses coverage at any point, regardless of the Number of Users. At the other end of the spectrum, the Greedy Algorithm's performance, regarding the QoS fluctuates greatly, but it is also apparent that in cases of large samples of users, it comes close to the Neighbor Awareness Algorithm. Same as before, the ideal conditions for the greedy algorithms are when the position of the users inside the network, is determined based on a Uniform Distribution.

Pairing the results of the two last Figures, we can conclude that in certain cases it is more beneficial to use the Greedy Algorithm and in other cases, we should employ the Neighbor Awareness Algorithm. For example, in places where the users are usually uniformly distributed, such as inside a household as stated in [12], the Greedy Algorithm could be put in use. In other cases where the uses are normally distributed, such as a city [12], where most residents reside in the center of the city, the Neighbor Awareness would prove to be more beneficial in terms of QoS and provide a relatively satisfactory amount of energy saved. As a conclusion from the above results, we can safely suggest that the Greedy Algorithm could be employed to femtocells, where the locations where they are deployed, mostly consist of homes, and medium office spaces, while the Neighbor Awareness Algorithm, should be used in conjunction with picocells.

V. Conclusions and future work

The implementation of Machine Learning approaches to enhance the ability to manage mobile networks in current and future trends and technologies was discussed in this paper. Our idea is presented and assessed with the goal of increasing the efficiency of the energy that is being expended inside the network. Moreover, solutions are offered that not only boost energy efficiency but also lower overall energy expenditure. To achieve this, we began by identifying a machine learning technique. This might be beneficial in the situation of existing networks, which employ a collection of cell-level traces that match to a real-world mobile network deployment. The approach in question in our scenario is classification, which is used to forecast the status of network components using real-time information and measurements that are being recorded by the BS and also processed locally, to produce the final result, which in our case is whether or not to shut down a cell.

ML Algorithms are then used in a simulation environment to evaluate alternative structures of an algorithm that conserves energy, which turns cells on and off depending on the network state. A future implementation of this mechanism could include further information and variables to train the model, as to make sure that no falsely classifications are being made and to increase the accuracy of the model. The implementation of the mechanism that decides which cells should be turned off or on, is done having two concepts in mind: how to save the largest amount of energy and how to save an acceptable amount of energy, while maintaining the QoS. Finally, these algorithms are investigated in scenarios that represent current and future, i.e., 5G network, deployments. The results show promise and reveal that a massive amount of energy can be saved on the part of the BSs. They also show that simply turning on and off small cells is insufficient to produce the necessary energy reduction. Switching off at the macro layer, on the other hand, might have a negative impact due to coverage gaps, thus service quality suffers.

VI. ACKNOWLEDGMENT

This research has been co-financed by the Hellenic Foundation for Research & Innovation (H.F.R.I) through the H.F.R.I.'s Research Projects to Support Faculty Members & Researchers (project code: 02440).

References

[1] Pihkola, H.; Hongisto, M.; Apilo, O.; Lasanen, M. Evaluating the Energy Consumption of Mobile Data Transfer—From Technology Development to Consumer Behaviour and Life Cycle Thinking. Sustainability 2018, 10, 2494.

[2] C. Han et al., "Green radio: radio techniques to enable energy-efficient wireless networks," in IEEE Communications Magazine, vol. 49, no. 6, pp. 46-54, June 2011, doi: 10.1109/MCOM.2011.5783984.

[3] Ericsson Mobility Report November 2020

[4] Langley, P. "The changing science of machine learning". Mach Learn 82, 275–279 (2011), doi.org: 10.1007/s10994-011-5242-y

[5] U. Challita, H. Ryden and H. Tullberg, "When Machine Learning Meets Wireless Cellular Networks: Deployment, Challenges, and Applications," in IEEE Communications Magazine, vol. 58, no. 6, pp. 12-18, June 2020, doi: 10.1109/MCOM.001.1900664.

[6] B. Matthiesen, A. Zappone, K. -L. Besser, E. A. Jorswieck and M. Debbah, "A Globally Optimal Energy-Efficient Power Control Framework and Its Efficient Implementation in Wireless Interference Networks," in IEEE Transactions on Signal Processing, vol. 68, pp. 3887-3902, 2020, doi: 10.1109/TSP.2020.3000328.

[7] I. AlQerm and B. Shihada, "Enhanced machine learning scheme for energy efficient resource allocation in 5G heterogeneous cloud radio access networks," 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, 2017, pp. 1-7, doi: 10.1109/PIMRC.2017.8292227.

[8] C. Bouras and G. Diles, "Energy efficiency in sleep mode for 5G femtocells," 2017 Wireless Days, Porto, 2017, pp. 143-145, doi: 10.1109/WD.2017.7918130.

[9] Niu, Zhisheng & Wu, Yiqun & Gong, Jie & Yang, Zexi. (2010). Cell Zooming for Cost-Efficient Green Cellular Networks. IEEE Communications Magazine. 48. 74-79. 10.1109/MCOM.2010.5621970.

[10] M. Moretti, L. Sanguinetti and X. Wang, "Resource Allocation for Power Minimization in the Downlink of THP-Based Spatial Multiplexing MIMO-OFDMA Systems," in IEEE Transactions on Vehicular Technology, vol. 64, no. 1, pp. 405-411, Jan. 2015, doi: 10.1109/TVT.2014.2320587.

[11] D. Lopez-Perez, X. Chu, A. V. Vasilakos and H. Claussen, "Power Minimization Based Resource Allocation for Interference Mitigation in OFDMA Femtocell Networks," in IEEE Journal on Selected Areas in Communications, vol. 32, no. 2, pp. 333-344, February 2014, doi: 10.1109/JSAC.2014.141213.

[12] I. Ashaf, F. Boccardi and L. Ho, "Sleep mode techniques for small cell deployments", IEEE Communications Magazine, vol. 49, no. 8, pp. 72-79, August 2011.

[13] H. Claussen, I. ashraf, and L. T. W. Ho, "Dynamic idle mode procedures for femtocells", Bell Labs Technical Journal, vol. 15, no. 2, pp. 95-116, September 2010.

[14] 3GPP TR 36.814 V9.0.0, "Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancemens for E-UTRA physical layer aspects (Release 9)", 3rd Generation Partnership Project, Tech. Rep., 2010.

[15] H. Claussen, "Perfomance of macro- and co-channel femtocells in a hierarchical cell structure", in Personal, Indoor and MobileRadio Communications, 2007, PIMRC 2007. IEEE 18th International Symposium on, Sept 2007, pp. 1-5.

[16] H. Lei, L. Zhang, X, Zhang and D. Yang, "A novel multi-cell OFDMA system structure using fractional frequency reuse", in Personal Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on, Sept 2007, pp. 1-5.

[17] P. Lee, T. Lee, J. Jeong, and J. Shin, "Interference management in LTE femtocell systems using Fractional Frequency Reuse", in 12th International Symposium on, Sept 2007, pp. 1-5.

[18] Bichler, Martin and Kiss, Christine, "A Comparison of Logistic Regression, k-Nearest Neighbor, and Decision Tree Induction for Campaign Management" (2004). AMCIS 2004 Proceedings. 230.

[19] Salazar, Diego Alejandro; Velez, Jorge Iván and Salazar, Juan Carlos. Comparison between SVM and Logistic Regression: Which One is Better to Discriminate?. Rev.Colomb.Estad. 2012, vol.35, n.spe2, pp.223-237. ISSN 0120-1751.

[20] M. Simsek, T. Akbudak, B. Zhao, and A. Czylwik, "An lte-femtocell dynamic system level simulator," in 2010 International ITG Workshop on Smart Antennas (WSA), Feb 2010, pp. 66–71.