



Article

eSCIFI: An Energy Saving Mechanism for WLANs Based on Machine Learning

Guilherme Henrique Apostolo ^{1,2,*} , Flavia Bernardini ¹ , Luiz C. Schara Magalhães ^{2,3} 
and Débora C. Muchaluat-Saade ^{1,2} 

¹ Institute of Computing, Universidade Federal Fluminense, Niterói 24210-240, Brazil; fbernardini@ic.uff.br (F.B.); debora@midia.com.uff.br (D.C.M.-S.)

² MídiaCom Lab, Universidade Federal Fluminense, Niterói 24210-240, Brazil; schara@telecom.uff.br

³ Telecommunications Engineering Department, Universidade Federal Fluminense, Niterói 24210-240, Brazil

* Correspondence: guilherme_apostolo@midia.com.uff.br

Abstract: As wireless local area networks grow in size to provide access to users, power consumption becomes an important issue. Power savings in a large-scale Wi-Fi network, with low impact to user service, is undoubtedly desired. In this work, we propose and evaluate the eSCIFI energy saving mechanism for Wireless Local Area Networks (WLANs). eSCIFI is an energy saving mechanism that uses machine learning algorithms as occupancy demand estimators. The eSCIFI mechanism is designed to cope with a broader range of WLANs, which includes Wi-Fi networks such as the Fluminense Federal University (UFF) SCIFI network. The eSCIFI can cope with WLANs that cannot acquire data in a real time manner and/or possess a limited CPU power. The eSCIFI design also includes two clustering algorithms, named cSCIFI and cSCIFI+, that help to guarantee the network's coverage. eSCIFI uses those network clusters and machine learning predictions as input features to an energy state decision algorithm that then decides which Access Points (AP) can be switched off during the day. To evaluate eSCIFI performance, we conducted several trace-driven simulations comparing the eSCIFI mechanism using both clustering algorithms with other energy saving mechanisms found in the literature using the UFF SCIFI network traces. The results showed that eSCIFI mechanism using the cSCIFI+ clustering algorithm achieves the best performance and that it can save up to 64.32% of the UFF SCIFI network energy without affecting the user coverage.

Keywords: WLAN energy saving mechanism; machine learning; RoD strategy mechanisms; smart buildings; Wi-Fi networks



Citation: Apostolo, G.H.; Bernardini, F.; Magalhães, L.C.S.; Muchaluat-Saade, D.C. eSCIFI: An Energy Saving Mechanism for WLANs Based on Machine Learning. *Energies* **2022**, *15*, 462. <https://doi.org/10.3390/en15020462>

Academic Editors: Shi-Jie Cao and Wei Feng

Received: 13 December 2021

Accepted: 6 January 2022

Published: 10 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The presence of Wireless Local Area Networks (WLANs) on shopping centers, convention centers, and commercial and university buildings has been increasing daily [1]. To cope with the increasing demand, new wireless Access Points (APs) are added to the network infrastructure in order to supply user demand with good Internet connection [2]. However, the deployment of new APs not only rises the network infrastructure maintenance cost, but also its operation costs [3]. These higher operation costs are mostly caused by energy consumption [4,5].

The energy consumption of large-scale wireless networks has raised concerns among researchers [3,4,6–9]. There are several studies in the literature proposing Resource on Demand (RoDs) strategies to improve the energy efficiency of those networks [7,10–14]. Wi-Fi RoD strategy management systems, or simply RoD strategy mechanisms, implement algorithms and policies that decide which APs should be switched off to save energy and which APs must stay switched on to cope with the traffic demands [1].

Some mechanisms use real time data acquisition or sophisticated RoD strategies that require great processing requirements to create their energy saving solutions [4,15,16].

However, there are wireless network scenarios where the Central Processing Unit (CPU) power is limited. This restriction makes the use of real-time occupancy's measurement and prediction impractical. Nevertheless, even those network scenarios could benefit from RoD strategy mechanisms and few to no adjustments would be required.

The Fluminense Federal University (UFF) SCIFI wireless network is a large-scale network developed by UFF, initially financed by RNP (Brazilian National Research and Education Network) [17]. The SCIFI network is a low-cost solution for large-scale wireless networks. Its implementation is open source and allows the control and management of those networks. The SCIFI network has two main components: The SCIFI smart controller; and the running APs, operating under the open source OpenWRT firmware [18].

It is possible to apply machine learning predictions to estimate APs occupancy demand in the UFF SCIFI wireless network scenario [19]. The key idea behind that is to use machine learning models predicted occupancy demand to control the power state of the APs during the day. The responsible for switching off APs according to the estimated demand for each time slot is the Wi-Fi network controller. The Wi-fi controller can do that by using RoD strategy mechanisms based on machine learning estimated demand. Machine Learning models are responsible for future occupancy demand estimations that our RoD mechanism bases its decisions on. Consequently, we need to conduct an analysis on the performance of machine learning models for our scenario. Some work use the Wi-Fi infrastructure to gather information about the Wi-Fi network occupancy history and use different classification and regression machine learning models to predict network usage [3,20–24]. However, to the best of our knowledge, none of them have investigated the combination of regression and classification predictions to improve the demand estimation accuracy or the combination of two RoD strategy algorithms to ensure client's association and the network minimum coverage for Wi-Fi networks.

This work proposes eSCIFI, an energy saving mechanism for WLAN. eSCIFI uses machine learning models to predict the wireless network future demand, therefore it can work in wireless networks where the controller's CPU power does not allow real time data acquisition to estimate this demand. eSCIFI uses two RoD strategy algorithms to ensure clients' association and the network minimum coverage: the AP clustering algorithm and the double threshold algorithm. The eSCIFI mechanism can determine which AP should be active or turned off during certain moments of the day in order to cope with the actual network demand and also save energy.

The main contributions of this work are:

- An energy saving mechanism named eSCIFI for WLAN, which can work in scenarios where real data acquisition is not possible;
- Analysis of how eSCIFI can cope with the network demand while saving energy and the comparison of its results with other RoD strategy mechanisms found in the literature. When compared to related work, eSCIFI achieved the highest energy saving factor (64.32%) without affecting the user access coverage.

The remainder of this work is organized as follows. Section 2 presents the related work to WLAN energy saving mechanisms. Section 3 describes the eSCIFI energy saving mechanism solution proposed in our work. Section 4 covers the evaluation of the proposed eSCIFI energy saving mechanism. Finally, Section 5 concludes this work, pointing out some enhancements and applications that might be explored in future work.

2. Related Work

Based on the work of Budzisz et al. [25], Jardosh et al. [2] and Lorincz et al [26], we developed an extended taxonomy that helped us to compare distinct RoD strategy mechanisms for WLANs. Our taxonomy consists of seven non-overlapping categories, corresponding to the main characteristics of related work: (1) network type, (2) WLAN application scenario, (3) control scheme, (4) operation strategy, (5) metrics, (6) algorithm type, and (7) evaluation method.

Most related work have developed RoD strategy mechanisms for Wi-Fi (IEEE 802.11) networks. However, there are great contributions in the literature that developed RoD strategy mechanisms for mesh [27] and cellular networks [8,24,28,29]. Those wireless network types have distinct characteristics, but the strategies and algorithms used on their RoD strategy mechanisms are interchangeable and sometimes even overlapping. It is important to notice that an RoD strategy mechanism developed and tested for a specific wireless network can be used in other wireless network types. Therefore, the network type category does not mean any sort of limitation to the RoD strategy mechanism applicability, but only describes the type of network used as the motivation and experimental scenario.

Most of the RoD strategy mechanisms were developed for application scenarios where they depend on homogeneous WLANs to operate, such as [2,7,11,12,14]. In those cases, the RoD strategy mechanism is implemented to fully cope with the WLAN technology without depending on any other wireless networks that might work in that area to help to implement their energy saving strategies. However, there are some RoD strategy mechanisms that were designed to operate in heterogeneous WLAN scenarios such as [9,30,31]. In the heterogeneous WLAN application scenarios, the WLAN can rely on other wireless technologies such as Bluetooth or in a separate wake-up radio transceiver to detect user activity while the WLAN infrastructure is turned off. The RoD strategy mechanism developed for heterogeneous WLAN application scenarios can usually achieve higher energy saving rates without affecting their user Quality of Service (QoS), since there is always a supportive wireless network to detect new users instantly. However homogeneous networks are less complex in terms of deployment, control and management, due to their independent WLAN nature.

The control scheme category expresses how the RoD strategy mechanism implements its energy saving strategy. The control scheme can be centralized or distributed. RoD strategy mechanisms with centralized control scheme uses a central controller to supervise the network and send the commands to APs. Centralized control schemes are more common for large wireless networks since most of them already have a central controller and their APs usually are not powerful enough to implement the algorithms and calculations needed. However, the centralized control scheme can be subdivided into two categories depending whether the central controller is designed for a Software Defined Network (SDN) or not.

SDNs separate the control and data plane by introducing a centralized controller that is responsible for resolving flows forwarding policies and assigning them to the switches' forwarding tables [10]. Some related work [10,14–16] developed energy saving mechanisms for SDN-based networks with a centralized SDN controller. The use of SDN controllers allows those energy saving mechanisms to use some collected network information such as network topology and traffic usage easily. However, not every large scale WLAN controller is based on the SDN paradigm and therefore can not count on all its advantages.

There are some proposed energy saving mechanisms in our related work that do not consider the controller to be SDN-based [3,8,27,32]. Those energy saving mechanisms also work with a centralized control scheme, but with non-SDN controllers which make them a feasible solution to WLANs where not all SDN advantages are present. On the other hand, in a distributed network, the WLAN elements are all responsible for controlling their energy state and deciding whether they can be turned off or not. However, it is important to highlight that a distributed control scheme does not necessarily mean that each WLAN AP works independently of the other. In [33,34], the Wi-Fi APs implement an energy saving strategy without a central controller, but they use out-of-band communication between them to decide which APs can be turned off.

RoD strategy mechanisms can be classified into two operation strategies: demand-driven or schedule-driven. Demand-driven strategies collect real-time information from the WLAN resources to estimate user demand [2]. The advantage of these strategies is that they can generate an energy saving in the WLAN while satisfying the user demand. However demand-driven strategies have a higher CPU power cost due to the overhead of assessing user demands continuously [6]. Demand-driven strategies are more suitable in

scenarios where the user demand may unpredictably vary over time such as in stadiums [7]. On the other hand, schedule-driven strategies use predefined schedules to produce its energy saving. These schedules can be obtained with machine learning models trained with WLAN historical usage data [3,12,29] or can be based on the administrator's experience [14]. The advantage of using schedule-driven strategies is their low CPU power requirements. Schedule-driven strategies are only suitable for scenarios where user demand is predictable, such as university networks [3,12,14].

The RoD strategy mechanisms can be divided into 4 metrics subsets according to the metrics they use to minimize the energy consumption. The most common and most intuitive metrics are the traffic metrics subset. The traffic metrics subset comprises any network traffic related metric such as number of associated users [3,12,32], throughput [8] or more sophisticated ones such as channel utilization [2]. Traffic metrics are often used and measured in a network, and therefore they are easily accessible, but they might not be enough to guarantee the QoS or coverage alone. Coverage metrics are used to ensure that the whole radio area network [2,6,14] and users [7,15,16] will be covered. Coverage implies that the RoD energy saving strategies will guarantee that all users can connect to at least one active radio. QoS metrics are often used in studies that try to minimize the impact on the user's service [13,28], but they also imply smaller savings or more complex algorithms to work. Energy metrics [4,26] consider the reducing energy quantitative for the analysis of switching on/off strategies. A clear implication is that the user's traffic or QoS constraints can not be met. One important thing to highlight is that every metric alone has its advantages and weaknesses, therefore most related work uses a combination of metrics to guarantee the user's demand will be met.

RoD strategy mechanisms can also be divided by the type of algorithm used for making the energy status decision for the WLAN resources based on the available metrics. Heuristic algorithms can rapidly determine a solution within reasonable time using reasonable resources [26]. As the name suggests, heuristic algorithms are based on heuristics solutions that are easier to implement and usually based on thresholds [6,12,29] or other metrics combination rules [30]. Heuristic algorithms are usually most suitable for WLAN scenarios where the CPU power and/or computational time required are low. On the other hand, optimization algorithms are based on different mathematical problems and solvers that guarantee the best possible solution to a specified problem [4]. Optimization algorithms require more time and resources to provide their solution and therefore are suitable for WLAN scenarios where the CPU power and/or computational time required are high. Related work show that optimization algorithms achieve better results when compared to heuristic ones [15,16], however, Lorincz et al. [26] concluded in their work that "heuristic algorithms can be valuable alternatives offering good solution in reasonable amount of time".

Lastly related work can be divided according to the experimental test made to evaluate their RoD strategy mechanism performance. Simulation tests are those that make use of simulation software such as Matlab [35], Scenargie [36] or NS-3 [14] to recreate their WLAN scenarios and evaluate performance. Trace-driven tests are those that use network traces to reproduce a real network scenario comparing how their network would respond to the changes in that scenario using distinct energy saving mechanism [3,12,24,29]. Testbed experiments are those where a real WLAN infrastructure is used, but a limited set of users and their behavior are simulated [1,7,37]. There are related work that refer to their tests as real network scenario tests [2,7], however, they do not analyze the real infrastructure in a regular usage scenario with undefined users or behaviors and therefore we classified them as testbed.

Table 1 compares related work to our proposed eSCIFI mechanism. It is important to highlight that eSCIFI can be used in a wider range of WLAN networks than most of the mechanisms presented in related work that have a centralized control scheme, since eSCIFI can cope with non-SDN-based wireless networks, does not have a high-CPU-power controller and cannot collect data in real time. Those characteristics make any energy saving

mechanism that presents optimization algorithms or demand-driven strategies unpractical. However, it is worth to mention that, in WLAN scenarios that present those characteristics, eSCIFI can work normally, but it might not be the best practical solution since it might not take advantage of those characteristics.

The eSCIFI characteristics make it a feasible solution for our motivation and evaluation scenario once it allows the development of an energy saving mechanism that can cope with the UFF SCIFI pure Wi-Fi network characteristics. eSCIFI presents a centralized controlling scheme, a schedule-driven operation strategy based on machine learning, using heuristic algorithms, traffic and coverage metrics.

Therefore, we can summarize the eSCIFI key contributions and advantages as:

- Introduces a centralized non-SDN-based control scheme;
- Proposes lightweight heuristic algorithms based on machine learning;
- Does not require real-time data acquisition;
- Does not require high-CPU-power controllers.

Table 1. RoD strategy mechanism related work comparison.

Related Work	Network Type	WLAN Scenario	Control Scheme	Operating Strategy	Metrics	Algorithm Type	Evaluation Method
Capone et al. [27]	Mesh Networks	Homogeneous	Centralized	Demand-driven	Coverage and QoS	Optimization	Simulation
Chen et al. [10]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	QoS	Optimization	Testbed
Chin et al. [11]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic and Coverage	Optimization	Simulation
Dalmasso et al. [8]	Cellular Networks	Homogeneous	Centralized	Demand-driven	Traffic	Heuristic	Trace-driven
Debele et al. [1]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic	Heuristic	Testbed
Donevski et al. [24]	Cellular Networks	Homogeneous	Centralized	Schedule-driven	Traffic	Heuristic	Trace-driven
Fang et al. [12]	IEEE 802.11	Homogeneous	Centralized	Schedule-driven	Traffic	Heuristic	Trace-driven
Ganji et al. [35]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Coverage	Optimization	Simulation
Gomez et al. [7]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic and QoS	Heuristic	Testbed
Jardosh et al. [6]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic and Coverage	Heuristic	Simulation
Jardosh et al. [2]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic and Coverage	Heuristic	Testbed
Kumazoe et al. [33,34]	IEEE 802.11	Homogeneous	Distributed	Demand-driven	Traffic and QoS	Heuristic	Simulation
Lee et al. [13]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Coverage and QoS	Optimization	Simulation and Testbed
Liu et al. [28]	Cellular Networks	Homogeneous	Centralized	Demand-driven	QoS	Optimization	Simulation
Lorincz et al. [4]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Energy and Traffic	Optimization	Simulation
Lorincz et al. [26]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Energy and Traffic	Heuristic	Simulation
Lyu et al. [3]	IEEE 802.11	Homogeneous	Centralized	Schedule-driven	Traffic	Heuristic	Trace-driven
Marsan et al. [32]	IEEE 802.11	Homogeneous	Centralized	Schedule-driven	Traffic	Heuristic	Trace-driven
Nagareda et al. [36]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic and Coverage	Heuristic	Simulation
Rossi et al. [9]	IEEE 802.11	Heterogeneous	Distributed	Demand-driven	Traffic and Coverage	Heuristic	Simulation and Testbed
Silva et al. [14]	IEEE 802.11	Homogeneous	Centralized	Schedule-driven	Traffic and Coverage	Heuristic	Simulation
Tanaka et al. [30]	IEEE 802.11	Heterogeneous	Centralized	Demand-driven	Traffic	Heuristic	Simulation
Vallero et al. [29]	Cellular Networks	Homogeneous	Centralized	Schedule-driven	Traffic	Heuristic	Trace-driven
Wu et al. [38]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic	Optimization	Simulation
Xu et al. [15,16]	IEEE 802.11	Homogeneous	Centralized	Demand-driven	Traffic	Optimization	Simulation
Yaodong Zhang et al. [37]	IEEE 802.11	Heterogeneous	Centralized	Demand-driven	Traffic	Optimization	Testbed
Yomo et al. [31]	IEEE 802.11	Heterogeneous	Distributed	Demand-driven	Traffic	Heuristic	Simulation
eSCIFI	IEEE 802.11	Homogeneous	Centralized	Schedule-driven	Traffic and Coverage	Heuristic	Trace-driven

3. Proposed eSCIFI Mechanism

In our previous work [39], we created a dataset using real user data collected from a subset of APs of the UFF SCIFI network located in a specific building of the engineering campus (the H building). That dataset provides the occupancy estimations for the H building during a period of 6 months (from April 2018 to September 2018). These features makes our dataset one of the biggest, most recent [3,12,32], and the only publicly available to the best of our knowledge. From the occupancy analysis [39], it was possible to observe that most network APs at the H building are switched on despite being idle. That active idleness causes an unnecessary waste of energy. Therefore, an energy saving WLAN mechanism based on RoD strategies, or simply RoD strategy mechanisms, that effectively controls WLAN resources can help to prevent those energy waste while coping with the user demand.

This work proposes the eSCIFI energy saving mechanism for WLANs. eSCIFI uses machine learning prediction models and other RoD strategies to create an energy saving mechanism. eSCIFI can also work with non-SDN large wireless networks and/or large wireless networks where real-time data acquisition is not possible. Those possibilities make the eSCIFI a feasible solution for a greater number of wireless networks in use, especially

university networks, such as the UFF SCIFI network, which was used for evaluating our proposal.

3.1. eScifi Mechanism Overview

Figure 1 shows eSCIFI main architectural components and its major steps, which are (i) the unified methodology; (ii) the hybrid model; (iii) heuristic algorithm.

The first step, shown in the left upper part of the figure, is to use our unified methodology to create the datasets and select the best regression and classification model configuration parameters. Later on, in the hybrid model, we combine the best trained regression and classification models selected in our unified methodology to give the future AP occupancy estimation. That occupancy estimation is used by our heuristic algorithm to define which APs should be turned on or off.

In the heuristic mechanism, we first extract the AP statistics from the dataset. Later on, the heuristic network clusters formation uses the AP neighborhood list and the AP statistics to create the network clusters that can guarantee a minimum network coverage. Finally, the energy state decision algorithm uses the defined network clusters and the AP occupancy estimation to decide which APs should be switched on/off to cope with the user demand. At the end of this process, our heuristic mechanism provides an energy scheduling of all APs in the network for an entire day that can guarantee a minimum coverage to the network while coping with the user demand. That way the eSCIFI mechanism needs to run only once a day to generate the energy scheduling of all APs in the network. Therefore, the eSCIFI mechanism can run at any moment of low activity in the network such as late night hours after midnight in our case. This scheme guarantees that eSCIFI can run at any network controller without burdening its processing capacity.

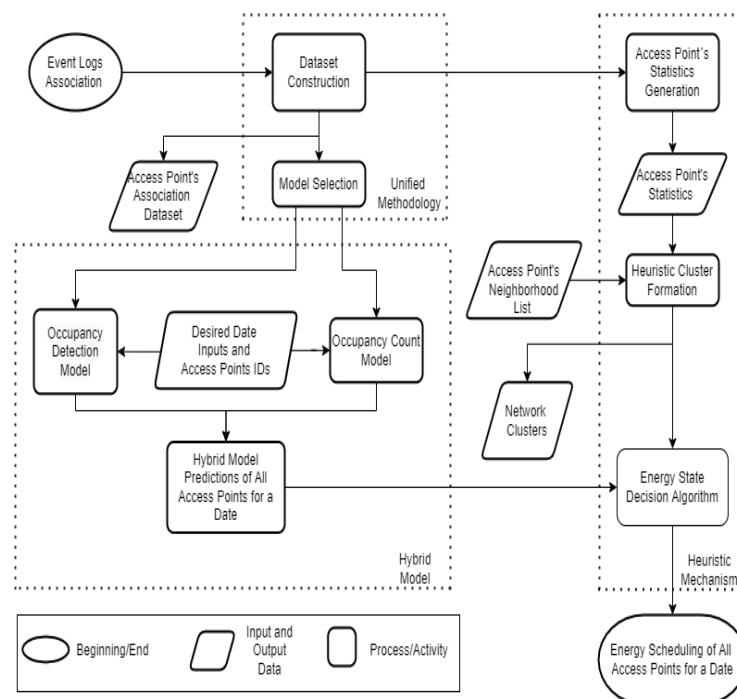


Figure 1. eSCIFI architecture.

3.2. Unified Methodology and Model Selection

The unified methodology proposed in [39] explains how the occupancy count (the amount of devices connected to an AP) and occupancy detection (if the AP is occupied or not) datasets were created. In summary, we have processed AP event logs to filter information about the association status between mobile stations and APs. Each day was divided into 144 time slots (10 min each), and for each time slot the number of associated devices was processed. This was computed for all the APs involved. The datasets show

occupancy count and detection of 28 APs in a classroom building at UFF's Engineering Campus over a period of 6 months, from April to September 2018. Those datasets were crucial to extract the AP's statistics that were necessary for the network clusters formation. They were also crucial to the model selection process.

The model selection process in our unified methodology compares several model configuration and hyperparameters in order to determine the best classification and regression models for our evaluation scenario. The evaluation involved the use of multiple classification (for occupancy detection) and regression (for occupancy count) models using a variety of configurations and algorithms. The following algorithms were used as classification models: Decision Tree (DT); K-NN; Random Forest (RF); and MultiLayer Perceptron neural network. As regression models, we used DT, K-NN, RF, XG optimized gradient boosting, support vector machine (SVM), stochastic gradient descent (SGD) algorithms and MultiLayer Perceptron neural network.

To evaluate these models, we applied a train/test split on our data where the dataset's association data from April to August were used for training, and the dataset's association data for the month of September were used for testing. We used several metrics to evaluate our classification (such as Accuracy and F-1 score) and regression (such as root mean square error and mean average percentage error) models performance. From the results shown in [39], it was possible to select the best classification and regression model for our scenario.

Results showed that the best classification and regression models were a single-label regressor and classifier trained using the decision tree algorithm in a collective manner where only one classifier and regressor were trained to predict the occupancy of all APs based on the previous data of all APs. The output feature is the occupancy estimation for the specific time slot. The models used the following attributes: Month, Day, Day of the Week, Holiday, Access Point Id, Hour, and Minute.

Therefore only one classifier and one regressor is needed for our scenario. Those classification and regression models used the decision tree machine learning algorithm and three input features (AP identification, day of the week and holiday). AP identification (APid) carries the access point identification number, day of the week indicates the respective week day and holiday indicates if the day is a day with lectures or not. Those are the machine learning classification and regression models selected and they will be used on the hybrid model to provide future usage predictions for the H building UFF SCIFI Wi-Fi network in our evaluation.

3.3. Hybrid Model

The results in [39] showed that even the best regression model has significant Root Mean Squared Percentage Error for a specific time slot t_j ($RMSPE_{t_j}$) values during night and morning time slots, but the $RMSPE_{t_j}$ values for time slots after midday decrease. On the other hand, the best classification model has relatively higher accuracy for a specific time slot t_j (A_{t_j}) values for night and morning time slots than for the rest of the day. Therefore we propose a hybrid model. The hybrid model combines the accuracy results given by the classification models with the regression results given by the regression models in order to create a better occupancy count estimation. Considering CMR as the classification results matrix that shows the occupancy detection estimations provided by the classifier for the APs and RMR as the regression results matrix that shows the occupancy count estimations provided by the regressor, we can define that the hybrid model estimation HMR is the Hadamard product result between both CMR and RMR matrices. Equation (1) shows the Hadamard product that produces the hybrid model results matrix that is used as the demand estimation by our mechanism.

$$HMR = CMR \circ RMR \quad (1)$$

Figure 2 shows how the hybrid model demand prediction results are closer to the real demand than the regression model demand predictions for the month of September 2018. In fact, Figure 2 shows that the hybrid model results can reduce the over demand

prediction that happened on the weekends (September 1, 2, 8, 9, 15, 16, 22, 23, 29, 30) and on the Brazil's Independence day public holiday (September 7).

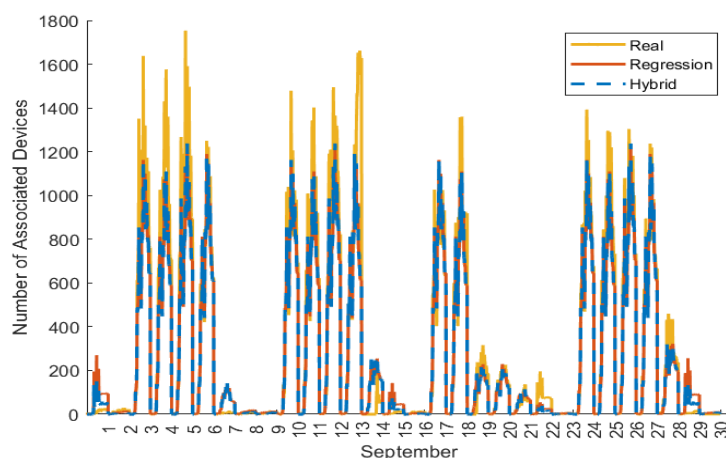


Figure 2. Hybrid model results compared with the real demand and the demand given by the regression results for the whole month of September.

The Hybrid model created only uses the APid, day of the week and holiday attributes as input features. Consequently, there are only 14 possible demand estimations for a specific AP (one for each regular day of the week and one for each holiday on these days). Therefore, we decided to compare the results of our hybrid model with a mean estimator. The occupancy count prediction provided by the mean estimator for a specific set of input features (APid, day of the week and holiday) is the average occupancy count of that specific set of input features in the association history. We compared the results of this mean estimator with the results of our hybrid model. Table 2 shows that the hybrid model had better Root Mean Squared Error (\overline{RMSE}), overall Root Mean Percentage Error ($\overline{RMSP\bar{E}}$) and overall Mean Absolute Percentage Error (\overline{MAPE}) results when compared to the mean estimator model. Those better results shown in Table 2 can be explained by the fact that the hybrid model has reduced the error predictions that happened on weekends and on public holidays when compared to the mean estimation results. Those reduced demands on weekend and on public holidays were more significant than the errors caused in night time slots by the hybrid model results, and therefore the overall \overline{RMSE} , $\overline{RMSP\bar{E}}$ and \overline{MAPE} results were better.

It is important to highlight that the difference between the results shown in Table 2 are not significant enough to prove that the hybrid model is a better regression prediction model than the pure regression model or mean estimator for all scenarios. The mean estimator results in our case scenario are very close to those achieved by the hybrid model. However, those results achieved by the mean estimator for our case scenario were only possible due the H building occupancy characteristics. The H building has only classrooms, so its occupation mainly occurs through lectures and exam applications. The lecture's schedule did not change drastically throughout the entire dataset which makes the occupancy behavior periodical and well behaved in our case. This behavior might not be common for other buildings in the university that have other room types, such as professor's offices or laboratories, or even other scenarios such as parks or malls. On other scenarios similar to ours, the mean estimator can be a viable option due to its simplicity. The use of the mean estimator does not impose any change to the eSCIFI operation. However, we decided to use the hybrid model since it has shown better results in our case scenario, specifically on weekends and holidays.

Table 2. Mean Estimator and Hybrid models performance results.

Metrics	Mean	Hybrid
\overline{RMSE}	8.4161	8.3996
\overline{RMSPE}	0.2977	0.2968
\overline{MAPE}	0.4189	0.4096

3.4. Heuristic Mechanism

The heuristic mechanism is responsible for providing the SCIFI APs energy state (on or off) schedule for a date. It is important to highlight that we only control the APs wireless interface energy state due to UFF SCIFI existing infrastructure that only allows us to control its energy state. However, in WLANs where the APs are connected to Power over Ethernet (PoE) switches, eSCIFI could normally control the energy state of the AP and not only its wireless interface.

Our heuristic mechanism has two main components: the heuristic cluster formation algorithm and the energy state decision algorithm. The clustering algorithm creates the AP clusters based on their neighborhood in order to guarantee the network coverage area to the clients. The energy state decision algorithm provides the energy state of all APs for a specific time slot and date based on the machine learning occupancy predictions and clusters. In the following sections, we detail the heuristic cluster formation algorithm and the energy state decision algorithm and its challenges.

3.4.1. Heuristic Cluster Formation: cSCIFI and cSCIFI+

Jardsoh et al. [2] proposed a clustering algorithm called green clustering. The idea behind the green clustering algorithm is to create clusters of APs that are in proximity of each other. Several APs in a large wireless network have overlapping coverage areas in order to cope with higher user demand. Those APs are in a spatially neighboring condition that allows one of them to provide coverage to the users of all APs in its vicinity. Therefore it is possible to create clusters of neighboring APs where any user within the cluster coverage is able to connect to the network as long as at least one AP in the cluster is turned on. We propose two heuristic cluster formation algorithms, cSCIFI (cluster SCIFI) and cSCIFI+ (cluster SCIFI+). Those clustering algorithms are based on the green clustering algorithm of Jardsoh et al. [2]. However, we introduced some basic changes to improve the cSCIFI and cSCIFI+ clustering formation process.

Our clustering algorithms need two input features to work: the neighborhood list and the AP statistics. To create a neighborhood list, we need to define the vicinity criteria. Only APs that are considered neighbors can belong to the same cluster. Jardsoh et al. [2,6] have used the spatial distance between APs and the median number of beacon messages and the median signal strength of the beacons as vicinity criteria. In our cSCIFI and cSCIFI+ algorithms, we are going to use the APs' signal quality scan to define our vicinity criterion. The SCIFI network periodically runs a signal quality scan that informs the different signal quality values received from the other APs that a certain AP has scanned. The signal quality is a measurement that goes from 0 to 100 and takes into consideration the Received Signal Strength Indication (RSSI) and other network parameters. We considered APs with a measured signal quality above 50 to be neighbors. Therefore, the neighbors of an AP are: (i) the APs on the same side of the building and floor; (ii) the APs that are in rooms directly above and below (e.g., neighbors to the AP in room 303 are the APs in rooms 403 and 203). With the established vicinity criteria, we can determine which APs are neighbors and create a neighborhood set list for each AP.

We describe our cluster formation algorithms as follows. Consider V_i as the neighborhood set of AP_i , C as our cluster set, and C_i as the cluster formed starting from AP_i . In our cSCIFI clustering algorithm, we first start by selecting AP_s with the biggest neighborhood set, forming a new cluster C_s and adding AP_s to its newly formed cluster C_s . When AP_s is added to its cluster C_s , the cSCIFI algorithm also removes AP_s from all other AP neigh-

neighborhood sets and update their number of neighbors. Then, the algorithm steps through all the APs in AP_s neighborhood set V_s and adds AP_h that has the biggest neighborhood set, as long as every new AP_h added to C_s is in the neighborhood set of all other APs already included in C_s . We call this the neighboring condition. As long as AP_s has APs on its neighborhood set V_s that satisfy the neighboring condition, those APs are added to cluster C_s and removed from the other AP neighborhood sets.

When there are no more APs in the AP_s neighborhood set or there are no more APs that satisfy the neighboring condition, the algorithm moves to the next AP with the biggest neighbor set and continues the cluster formation until there are no more APs left and the cluster set C is finished.

Algorithm 1 shows the cSCIFI cluster function code, where we can see that every AP will only be in one cluster and that every AP is on the vicinity of all other APs inside its cluster. The neighboring condition (line 5) allows any user in the cluster coverage area to connect to any of the powered-on APs , since they are all each other's neighbors.

Algorithm 1 cSCIFI

```

1: function Create_Cluster_cSCIFI (Cluster_Head, Cluster_head_list_of_neighbors):
2:   Cluster_auxiliary_list = [Cluster_Head]
3:   sort  $APs$  in Cluster_head_list_of_neighbors according with the number of
   neighbors in their neighborhood list
4:   for  $AP$  in Cluster_head_list_of_neighbors:
5:     if  $AP$  in neighborhood list of all Cluster_auxiliary_list elements:
6:       add  $AP$  to Cluster_auxiliary_list
7:       remove  $AP$  from neighborhood list of all  $APs$ 
8:   return Cluster_auxiliary_list

```

cSCIFI+ is simpler and more aggressive than cSCIFI. The cSCIFI+ clustering algorithm works like the cSCIFI, but now the APs added to a certain cluster C_i do not need to cope with the neighboring condition. In cSCIFI+, all neighbors in the AP_A neighborhood AP set V_A are added to cluster C_A .

cSCIFI+ guarantees that the size of cluster set C will be the smallest possible. However, users from a switched-off AP in the cluster can only connect to AP_A that initiated that cluster. Considering the clusters formed with cSCIFI, users from any AP can connect to other APs in that cluster, which might balance the load between the switched-on APs . Algorithm 2 shows the cSCIFI+ cluster function code, where we can see that only the AP that initiated the cluster formation can assure connection to all users from switched-off APs , which may sometimes cause congestion.

Algorithm 2 cSCIFI+

```

1: function Create_Cluster_cSCIFI+ (Cluster_Head, Cluster_head_list_of_neighbors):
2:   Cluster_auxiliary_list = [Cluster_Head]
3:   for  $AP$  in Cluster_head_list_of_neighbors:
4:     add  $AP$  to Cluster_auxiliary_list
5:     remove  $AP$  from neighborhood list of all  $APs$ 
6:   return Cluster_auxiliary_list

```

The cSCIFI and cSCIFI+ greedy algorithms alone cannot guarantee that the best cluster set is formed in cases where there is a tie between APs . A solution would be creating all cluster possibilities, choosing each one of the tied APs as the first choice. After creating all possible sets C , we would select the one that has the minimum number of clusters. Those multiple cluster sets creation can cause an exponential growth in the execution time. Trying to minimize those problems, we simplified the cSCIFI and cSCIFI+ selection in cases of ties. The cSCIFI and cSCIFI+ will only create multiple cluster sets when there are ties between

APs that will be selected to initiate a cluster formation. This selection criteria will guarantee that only different clusters initiation will be taken into relevance and not all possible cluster internal formations, which will minimize the possible solution set.

In the cSCIFI algorithm, we also added another selection criterion for cases where there are ties between *APs* to be added to cluster C_i where an AP_i has already initiated it. In those cases, AP_j with the highest number of neighbors in the AP_i neighborhood set V_i is selected. *APs* with the same number of neighbors in their sets can generate different clusters, since some of their neighbors might not be in AP_i neighborhood set V_i . Therefore, in cases of ties, it is the best option to select AP_j that has the biggest number of matching neighbors to the *APs* in the neighborhood set V_i . This change in the internal cluster formation process guarantees that the next *APs* to be added will be the ones that will contribute to a bigger cluster size.

Those characteristics cited previously minimizes the execution time and guarantee that a possible cluster set C will be selected independent of their appearances on the cluster neighborhood list. This is an important advantage to our clustering algorithms when compared to the green clustering algorithm proposed by Jardosh et al. [2], since we do not need to worry about the *AP* order of appearance in the neighborhood list construction process.

The last characteristic of our clustering algorithms is the cluster head election. The cluster head is the *AP* that will be always switched on and will be responsible for guaranteeing the cluster coverage area. In cSCIFI+, the cluster head will always be the one that initiated the cluster formation. This *AP* is the only *AP* that can be the cluster head, since this is the only *AP* that has a guaranteed neighboring condition to all other *APs* in the cluster. On the other hand, the election of the clusters head in the cSCIFI algorithm can be more sophisticated since all *APs* in the clusters obey the neighboring condition. In clusters formed with cSCIFI, the cluster head will change throughout the day using the statistics of the *APs*. In those clusters, the average association of each *AP* is calculated for the night (0 a.m.–7 a.m.), morning (7 a.m.–1 p.m.) and afternoon/evening (1 p.m.–11 p.m. 59') periods. The *AP* with the highest night average association will be selected to be the cluster head for the night period and so on.

3.4.2. Energy State Decision Algorithm

The energy state decision algorithm is responsible for providing the energy scheduling of all *APs* for a date. It runs once a day and uses the traffic demand estimated by the hybrid machine learning model (the user association number in our SCIFI network scenario) to calculate the cluster demand for specific moments of a day and then decide which *APs* in a cluster can be switched off. The energy state decision algorithm is the last step on the eSCIFI energy saving mechanism, and it is responsible for actively deciding which *APs* will be switched on or off and to provide the energy scheduling to the SCIFI controller. The SCIFI controller, based on this energy scheduling, will then control the *AP* wireless interface switching on and off for the specified periods.

Our energy state decision algorithm uses the RoD policy proposed in the work of Dalmaso et al. [8]. However, our energy state decision algorithm works using machine learning occupancy estimations instead of real traffic data and therefore presents some modification in the RoD policy design. This RoD policy has two main components: the time window and the double threshold criteria. The time window defines how long it will take before the algorithm reconfigures the *AP's* energy state. The time window size tw informs on which frequency the network will be reconfigured and also the demand estimation resolution. A small time window will allow the energy state decision algorithm to perceive short bursts in the traffic demand variations. On the other hand, a large time window will only perceive the average traffic where instant or momentarily bursts in the traffic demand will fade. At first, a smaller time window size seems always the best choice, however a smaller time window size means more rounds of energy state decisions will have to be made by the algorithm and that the controller will have to reconfigure the network more

frequently. Related work [3,8,12,14,24] state that small time window sizes are not necessary. In fact, depending on the network traffic profile, those changes in the traffic demand can take hours to happen. Therefore, the selection of the time window size is a parameter that needs to be decided based on the network scenario. In Section 4.1, we will deeply discuss the selection of the time window size.

The main concept behind our energy saving strategy is moving the traffic demand from switched off APs to the cluster head AP or other switched on APs in the cluster that can handle them. In the work of Damalso et al. [8], the APs in a cluster can be switched off based on the actual traffic demand (real time traffic data) at the beginning of each time window. However, the eSCIFI mechanism uses machine learning models to estimate demand. Therefore, in our energy decision algorithm, the decisions made for each time window will take into consideration the demand estimated for its whole duration and not just the demand at the beginning of the time window.

All APs in the network have the same maximum user threshold T_{max} for a time window. This maximum user threshold T_{max} defines how much traffic (or how much associations in our case) the APs can handle for the duration of the time window. The cluster head of every cluster will always be switched on guaranteeing a traffic capacity of T_{max} for the cluster. In our energy state decision algorithm, the double threshold criteria defines which APs in a cluster can be switched off based on the traffic demand estimated by the machine learning hybrid model for the assessed time window. However, this energy state decision algorithm varies depending whether cSCIFI or cSCIFI+ algorithms are used.

In the cSCIFI algorithm, all APs in a cluster are neighbors between themselves. Therefore, in the cSCIFI case, the double threshold criteria defines that APs with estimated traffic demand below a minimum threshold T_{min} for the whole time window are switched off as long as the available traffic capacity provided by all APs that are switched on can handle their estimated traffic. Considering DM_i as the traffic demand of AP_i for a time window, d as the number of switched on APs, o as the number of switched off APs, $\sum_{a=1}^d DM_a$ as the traffic demand of all d switched on APs and $\sum_{a=1}^o DM_a$ as the traffic demand of all o switched off APs, we can define how our energy state decision algorithm decides if an AP_i will be switched off based on the double threshold criteria if the cSCIFI algorithm is used. Equation (2) shows the double threshold criteria, where the first criterion defines if the traffic demand is too low for the AP_i to be switched on and the second criterion defines if the cluster switched-on APs can handle the AP_i traffic. If there are more d switched-on APs in the cluster, the cluster maximum traffic capacity CCA increases to $(d + 1)T_{max}$ because cSCIFI guarantees that all APs inside a cluster can provide connection to any mobile station trying to connect to any AP in the cluster.

$$\begin{cases} DM_i < T_{min} \\ CCA - (\sum_{a=1}^d DM_a + \sum_{b=1}^o DM_b) \geq DM_i, & \text{where} \\ CCA = (d + 1)T_{max} \end{cases} \quad (2)$$

On the other hand, in cSCIFI+, all APs in a cluster are neighbors only of the cluster head. Therefore, APs with estimated traffic demand below a minimum threshold T_{min} for the whole time window are switched off as long as the available traffic capacity provided by the cluster head can handle their estimated traffic. Equation (3) shows the double threshold criteria if cSCIFI+ is used, where the first criterion defines if the traffic demand is too low for AP_i to be switched on and the second criterion defines if the cluster head can handle the AP_i traffic. In cSCIFI+, the cluster maximum traffic capacity CCA is fixed to T_{max} because cSCIFI+ guarantees that only the cluster head can provide connection to any mobile station trying to connect to any AP in the cluster, except the AP itself.

$$\begin{cases} DM_i < T_{min} \\ CCA - (\sum_{a=1}^d DM_a + \sum_{b=1}^o DM_b) \geq DM_i, & \text{where} \\ CCA = T_{max} \end{cases} \quad (3)$$

eSCIFI has several parameters that must be configured and that may depend on the network usage profile, such as the selection of the time window size and T_{min} value. In the next section, we evaluate how the different components in the eSCIFI architecture affect the mechanism energy saving capacity and the network coverage to its users. We also compare eSCIFI to other related work about energy saving mechanisms that are applicable in our evaluation scenario.

4. eSCIFI Evaluation

To evaluate how eSCIFI impacts on the network performance, we performed trace-driven simulations using the real association trace data collected from the UFF SCIFI network. Our trace-driven tests use UFF SCIFI's association traces to reproduce a real network scenario. The idea is to compare how the network would respond to the changes in that scenario using distinct energy saving mechanisms. A trace-driven test does not require using a network simulator as NS-3 for example. It allows estimating the metrics by simply inputting the real association traces to the eSCIFI mechanism and then evaluating if eSCIFI can cope with user demand while saving energy. To perform our simulations, we are going to use the association data collected for one week in September 2018 from the H building at UFF. The week used in our collected data is formed by a weekend (1 and 2 September 2018) and 5 weekdays from Monday to Friday (24–28 September 2018). The weekends used are apart from the weekday dates because there were not complete association history traces for the weekend before or after those weekdays. This might have happened for several reasons such as energy outages or network failures for example. However, the weekend (1 and 2 September 2018) contains the association data for all APs in the H building, and therefore will be used to represent Saturday and Sunday in our trace-driven simulations. We are also going to use the Brazil's Independence day public holiday (September 7) to compare and evaluate how eSCIFI impacts the network on holidays.

The work of [40] presents a mathematical formula, indicated in Equation (4), that allows us to determine the energy saving factor ESF achieved with the AP wireless network interface shut down during periods of time. The formula gives the saved energy percentage when shutting down the AP wireless network interface compared to the total energy that would be consumed if its interface works the whole time.

Terms P_{ext_on} and P_{ext_off} of Equation (4) represent the measured power values in Watts, in the AP external power source, when the wireless network interface is powered on and off, respectively. Terms t_{on} and t_{total} represent the amount of time the AP stayed with its wireless interface switched on and the total analysis elapsed time, respectively. Equation (4) provides the percentage of energy that could have been saved by switching off the wireless interfaces of the AP during the idle time slots. This formula can be easily extended to also provide the network's energy saving factor. To do so, the terms t_{on} and t_{total} must change in order to represent the sum of time that all APs on the network and the total time multiplied by the number of APs in the network, respectively.

$$ESF = \frac{P_{ext_on} - P_{ext_off}}{P_{ext_on}} \left(1 - \frac{t_{on}}{t_{total}}\right) \quad (4)$$

From Equation (4), it is possible to notice that ESF reaches its maximum power saving factor value, ESF_{max} , when $t_{on} = 0$. This condition represents the scenario where the wireless interface of all APs in the network are switched off during the whole time. However, it is also possible to notice that, depending on the scenario and switching off scheme, ESF_{max} can assume several values. Therefore, the normalized energy saving factor, \overline{ESF} given by Equation (5), can better indicate the performance of the mechanism in different scenarios. The normalized energy saving factor \overline{ESF} is limited between 0% and 100% and represents the percentage of the maximum energy saving factor that could be saved.

$$\overline{ESF}(\%) = \frac{ESF(\%)}{ESF_{max}(\%)} \quad (5)$$

The work of [12] defines the coverage ratio loss CR formula, indicated in Equation (6). The coverage ratio loss is the number of uncovered clients U_l by the energy saving mechanism over the total clients in the network U within a certain period of time. The coverage ratio loss gives the percentage of clients that could not successfully access the network under the evaluated period.

$$CR(\%) = \left(\frac{U_l}{U} \times 100 \right) \quad (6)$$

The analysis in this section will evaluate the normalized energy saving factor (Equation (5)) and the coverage ratio loss (Equation (6)) to compare how the eSCIFI mechanism impacts on the network performance. To calculate the coverage ratio loss, we must know the parameter T_{max} that indicates the maximum number of association an AP might support in a time slot. We defined $T_{max} = 300$, which is roughly the maximum number of AP s associated in a time slot registered plus 10%. In our experimental scenario, only the wireless network interface will be switched off. Table 3 shows the consumed power measured for the AP model present in the UFF SCIFI network when the wireless interface is switched on and off (P_{ext_on} and P_{ext_off}). Table 3 also shows what would be the maximum power saving factor, ESF_{max} , which represents the power saving factor percentage if the wireless interface of all AP s were switched off the whole time. Therefore, in our evaluation scenario the maximum energy saving factor percentage that could be reached by switching off the wireless interface of the entire network during the whole evaluation period is 23,93%. That information is required by the normalized energy saving factor calculations.

Table 3. AP 's consumed power and maximum power saving factor percentage.

P_{ext_on} (W)	P_{ext_off} (W)	ESF_{max} (%)
1.111	0.845	23.93

We are going to evaluate how several components from the eSCIFI architecture impact on the network performance. eSCIFI using the cSCIFI and the cSCIFI+ clustering algorithms will also be compared with other mechanisms proposed in the literature. The eSCIFI energy saving mechanisms will be compared with SEAR, ACE and ECMA mechanisms proposed by Jardosh et al. [2], Fang et al. [12] and Silva et al. [14], respectively. The SEAR mechanism uses the green clustering algorithms and a single threshold where only the T_{min} parameter is used as the RoD strategy. In the SEAR mechanism, the network AP s are grouped into clusters, the cluster head is always switched on and the other AP s in the clusters remain switched off as long as their traffic demand is lower than T_{min} . The ACE mechanism uses an inactivity time window based on machine learning occupancy detection results as its RoD strategy and does not have a coverage guarantee. AP s that remain unused by a whole time window size are switched off the whole time window duration period. The ECMA mechanism uses the SEAR mechanism for night hours (between 0 a.m.–6:59 a.m.) and keeps the whole network switched on the rest of the day. The Baseline mechanism where all the AP s in the network remain switched on between 7 a.m.–11:59 a.m. and switched off between 0 a.m.–6:59 a.m. is also used for comparison.

We will evaluate how the time window size and the minimum threshold value affect the network performance. After that, we will also compare the eSCIFI mechanism performance on regular weekday with its performance on a public holiday. Our last analysis will compare the SEAR green clustering, eSCIFI with cSCIFI and cSCIFI+ clustering algorithms performances using different neighborhood lists. Our trace-driven simulation scripts were developed in Python.

4.1. Time Window Size Analysis

In Section 3.4.2, we have seen that the time window tw defines how long it takes before reconfiguring the network AP s energy state. A bigger time window is desired since

it will minimize the number of times the controller will need to change the APs working status, which will minimize the controller tasks over a day. On the other side, a bigger time window may not notice small traffic demand bursts, which may lead to network coverage losses during these bursts due to unnoticed behaviors. Therefore, we need to evaluate how the eSCIFI time window size may affect the network energy saving and coverage loss. We tested the eSCIFI mechanisms using 5 different time window values (10 min, 30 min, 1 h, 1:30 h and 2 h). Those time window values were selected based on our time slots size and correspond to 1, 3, 6, 9 and 12 time slots, respectively. Those time windows were selected based on the lecture duration time at UFF, which usually takes 2 h. The real and predicted association values for time windows bigger than one time slot (10 min) is the sum of the devices connected during the corresponding amount of time slots. We evaluated eSCIFI using the cSCIFI and cSCIFI+ clustering algorithms and $T_{min} = 72$ with different time windows to evaluate the normalized energy saving factor and coverage loss. Those fixed parameters were used because they delivered the best normalized energy saving factor percentage and coverage ratio loss to all possible time windows. We will also evaluate the time window size effect in the SEAR, ACE, ECMA and Baseline mechanisms.

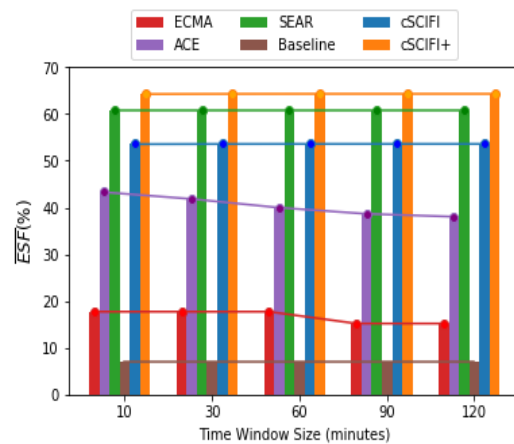


Figure 3. Normalized energy saving factor for different time window sizes.

As we can see in Figure 3, the selected time window sizes have not affected the normalized energy saving factor \overline{ESF} for SEAR and eSCIFI using the cSCIFI and the cSCIFI+ clustering algorithms. Only ECMA and ACE had their normalized energy saving factor negatively affected by the time window size. The baseline estimator does not depend on the time window (its scheduling presents fixed switching on/off periods), and therefore we can see that its normalized energy saving factor does not change. This result means that for our evaluation scenario it is possible to use a 2-h time window resolution without affecting the normalized energy saving factor for our eSCIFI mechanism. This would allow the eSCIFI mechanism to compute less energy state changes in the APs and consequently less tasks to be executed by the wireless network controller.

Figure 4 shows how the different time window sizes affects the coverage ratio. As we can see only Baseline and ACE presented coverage ratio losses in this evaluation scenario. The baseline estimator has a fixed coverage loss that does not depend on the time window size. The Baseline loss occurs due to unattended users in the night hours where all APs are switched off. However, the ACE mechanism shows a small decrease in the coverage loss as the time window grows. That result was expected because ACE uses the time window size as an inactivity criteria to switch off APs, and therefore a large time window would require a longer period of inactivity, which would be harder to achieve and consequently would lower the chances of mistakenly switching off APs.

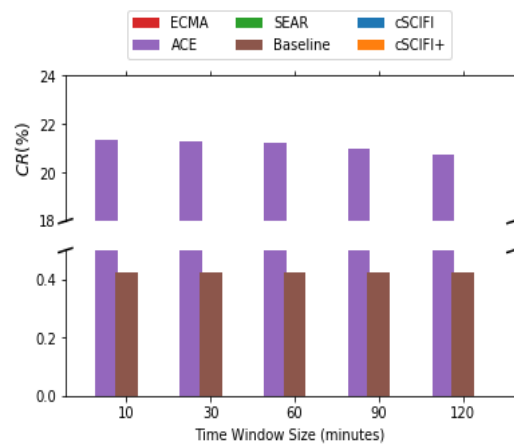


Figure 4. Coverage loss for different time window sizes.

4.2. Minimum Threshold Analysis

The last parameter on our eSCIFI mechanism that needs to be evaluated is the T_{min} value selection. T_{min} defines the minimum number of associations that an AP must have during the time window duration to be switched on. If the number of associations is below T_{min} , the AP will be evaluated to be switched off by the energy state decision algorithm. In this section, we evaluate how the T_{min} value affects the normalized energy saving factor and the coverage ratio. To do so, we varied the value assumed by T_{min} during one time slot including all multiples of 9 ranging from 9 to 90. Therefore, the T_{min} value will be proportional to the time window size used. Therefore, if the time window has a size w of time slots, the T_{min} values assumed will be $w \times T_{min}$. We fixed the time window size to 12 time slots (2 h or 120 min).

Figure 5 shows the normalized energy saving factor achieved by eSCIFI using the cSCIFI and cSCIFI+ clustering algorithms, SEAR, ACE and ECMA. As it can be seen in Figure 5, SEAR and eSCIFI using cSCIFI+ got the best energy saving percentages on our evaluation scenario. eSCIFI using cSCIFI had a smaller energy saving percentage because it has a different cluster set that is bigger than the ones formed by the SEAR eSCIFI using cSCIFI+. From Figure 5, we can also see that the normalized energy factor \overline{ESF} grows as T_{min} grows until it reaches $T_{min} = 54$, after that, the energy factor stays the same for all mechanisms. This result was expected and it is the same result achieved by Dalmaso et al. [8]. This asymptotic characteristic in the normalized energy saving factor curve happens because, for values of T_{min} higher than 54, the cluster maximum capacity CAA threshold is reached requiring those same APs to be turned on anyway.

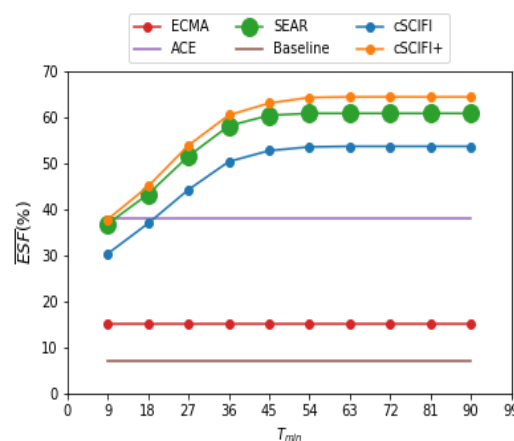


Figure 5. Normalized Energy saving factor for different T_{min} values.

Higher T_{min} values mean that APs will require a higher number of associations in a time window to be switched on according to the first criteria, which means it will be harder for them to be switched on. However, those APs will have their demand transferred to the cluster head (or other switched on AP in the case where cSCIFI has been used). That will mean that the cluster maximum capacity CAA threshold will be reached sooner and the APs will have to be turned on anyway. Therefore the normalized energy saving factor is limited and there is a T_{min} value that reaches it. Increasing T_{min} after its optimum value will not change the normalized energy saving factor. The possible explanations behind that may be that after $T_{min} = 54$, SEAR already reaches the minimum required APs to guarantee coverage (only the cluster heads may be switched on) or the switched on APs after that value present traffic demands much higher than the maximum value of $T_{min} = 90$. Figure 5 shows that ECMA has a steady normalized energy saving factor that does not depend on the T_{min} value. This might happen because ECMA applies the SEAR mechanism in night hours (between 0 a.m–6:59 a.m) and keeps the whole network switched on the rest of the day. The network has very little traffic demands in night hours, therefore few APs are required to be turned on or will have enough traffic to trigger the cluster maximum capacity threshold. That way, ECMA already reaches its highest energy saving factor with a $T_{min} = 9$. ACE and Baseline do not present a T_{min} parameter for energy state decision and therefore their normalized energy saving factors do not change.

We also evaluate how different T_{min} values affect the coverage ratio. As we can see in Figure 6, eSCIFI using both clustering algorithms (cSCIFI and cSCIFI+), SEAR and ECMA strategy had no coverage ratio loss at all for any value of T_{min} . This results showed that none of the mechanisms had overpassed the maximum cluster capacity at any moment. Only Baseline and ACE present coverage losses. However, as we have already mentioned previously, those mechanisms do not change their energy state decisions based on a minimum threshold T_{min} parameter. Therefore, their coverage ratio results are the same showed in Figure 4, where the time window size is $tw = 120$ min.

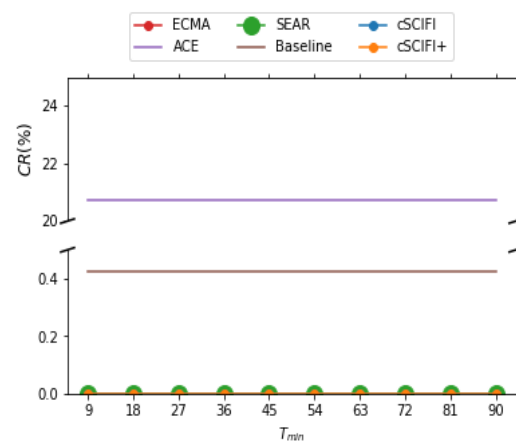


Figure 6. Coverage ratio loss for different T_{min} values.

4.3. Weekday Versus Holiday Analysis

The eSCIFI uses machine learning prediction models to estimate traffic demands. In our scenario, the hybrid model uses a holiday input feature that distinguishes normal weekdays from public holidays and university student holidays. The hybrid model uses this feature to differentiate the network demand variation that happens between regular day and holidays. Here, we will evaluate if eSCIFI using the hybrid model can better cope with the holiday demand than SEAR, ACE and ECMA. We compare Brazil's Independence Day public holiday (Friday, September 7) and the Friday used in our regular week. We compared the mechanism using the parameters that gave the best normalized energy saving factor and smallest coverage ratio loss ($T_{min} = 54$ and $tw = 120$ min).

Figure 7 shows the normalized energy saving factor achieved by the different mechanisms. As we can see, eSCIFI using both clustering algorithms and SEAR kept the normalized energy saving factor stable. eSCIFI using cSCIFI+ has the biggest normalized energy saving factor for holiday and weekdays. Baseline and ECMA also remain with their normalized energy saving factor unchanged. For Baseline, this happens because the decision is only based on time schedules and not on traffic demand estimations and therefore is unaffected. For ECMA, this result happened because the traffic demand for holiday or weekday remains unchanged, which did not change the SEAR APs switching on/off schedule during night hours. Only ACE had a reduction on the normalized energy saving factor in our holiday evaluation.

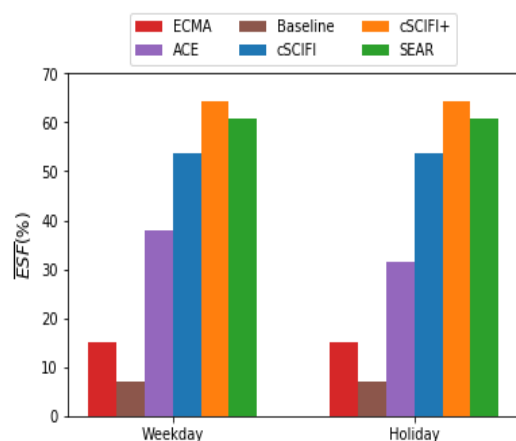


Figure 7. Normalized Energy saving factor comparison between a holiday and a weekday.

As we have seen in Figure 2, the demand on that holiday (Friday, September 7) was much smaller than the demand presented for the regular weekday (Friday, September 28). Figure 2 also shows that the holiday demand predicted by the hybrid model is much bigger than the real one, differently from the regular Friday where the hybrid model prediction was very close to the real traffic. Those results would first suggest that the normalized energy saving factor achieved by eSCIFI and SEAR for the public holiday should have been smaller as it happened with ACE. However, Figure 2 shows that the hybrid model wrong estimations have not even reached 200 associated devices for the whole network in any moment of the day on September 7. Figure 2 also shows that the regular Friday has not even reached 500 associated devices for the whole network on September 28. Those association values are very low considering that $T_{max} = 300$. Therefore, we can presume that, for the evaluated regular and holiday Friday, the network is working with the minimum set of APs switched on (only the cluster heads) and that is the reason why SEAR and eSCIFI using both clustering algorithms have their normalized energy saving factor unchanged. In fact, we analyzed how the real data would affect the normalized energy saving factor results in that analysis for the mechanisms and it showed that it would not have changed much (less than 3.3% for all mechanisms) in the results.

Figure 8 shows the mechanisms' coverage ratio loss for the regular weekday and for the holiday. Only ACE and Baseline present some energy loss since they are the only mechanisms that do not have a coverage guarantee. The Baseline coverage loss remains the same, which shows that the traffic demand for night hours (0 a.m.–6:59 a.m.) on both our holiday or weekday remains unchanged. The smaller coverage ratio loss for our holiday when compared to our weekday on the ACE mechanism case can be explained by the smaller traffic demand estimated for the whole day. Another explanation for the ACE mechanism reduced coverage ratio can be on the fact that the ACE mechanism has a smaller normalized energy saving factor on our holiday, which means it has a smaller number of APs switched off or that they are switched off for a short period of time.

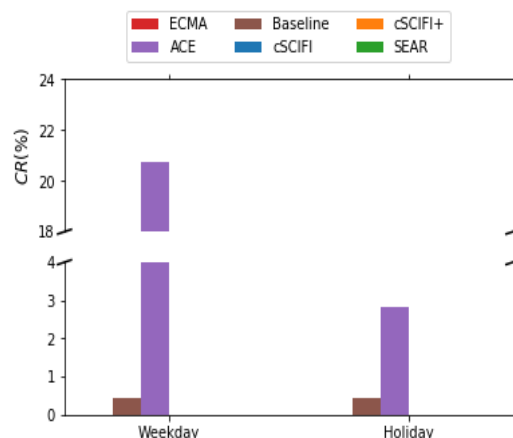


Figure 8. Coverage ratio loss comparison between a holiday and a weekday.

The results shown in this section cannot give us a precise conclusion on whether or not our algorithm could cope with the holiday demand without sacrificing the normalized energy saving factor. A large number of holidays in distinct weekdays and with distinct demand estimations would be necessary to understand it better. However, results indicated that the mechanism performance on holidays is not related to any change on its function, but it is in fact intimately related to the correct traffic estimations given by the hybrid model when compared to the real traffic data.

4.4. SEAR vs. eSCIFI Clustering Algorithms

As we have seen in Section 4.1, SEAR had a better normalized energy saving factor result than eSCIFI using the cSCIFI clustering algorithm. However, the clustering algorithm developed by Jardosh et al. [2] does not have the same optimizations criterion we have implemented on our both algorithms. Therefore, the work of Jardosh et al. [2] is susceptible to the order of appearance of *APs* in the neighborhood list of other *APs*. This order affects which will be the next *APs* selected by the Jardosh et al. [2] green clustering algorithm to fill the cluster in case of ties between the number of neighbors. The order of appearance of *APs* in the neighborhood list impacts its result since it does not have any tie breaker rule in the selection of the next *AP* to be put in the cluster in case of a tie in the number of neighbors between *APs*. cSCIFI and cSCIFI+ do not have this disadvantage, and therefore we can guarantee that the clusters formed will not depend on the order of appearance. Here, we will compare the normalized energy saving factor achieved by the SEAR and eSCIFI mechanism using both clustering algorithm using 3 different orders of appearance of *APs* in the neighborhood lists of the *APs*. The two first neighborhood lists present cases where the *APs* position inside the neighborhood lists are randomized and the third represent the neighborhood list we have used for all tests we have done before for the SEAR mechanism. We will compare the SEAR and eSCIFI mechanism using both clustering algorithms with the parameters that gave the best normalized energy saving factor and no coverage ratio loss ($T_{min} = 54$ and $tw = 120$).

As we can see in Figure 9, SEAR energy saving result is heavily affected by the order of appearance of *APs* in the neighborhood list, while eSCIFI using cSCIFI and cSCIFI+ are not affected at all. This result shows that the changes we have implemented on cSCIFI and cSCIFI+ have turned our algorithm unaffected by the order of appearance of *AP* in the neighborhood list. This is a clear advantage since it will not require an optimization on the neighborhood list formation process that in a huge network topology might be unpractical to be done.

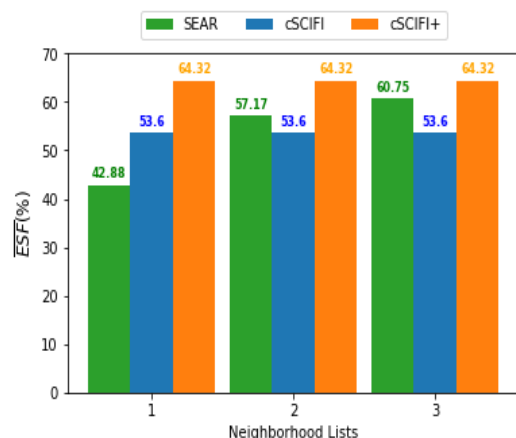


Figure 9. Normalized Energy saving factor comparison between SEAR and eSCIFI using both clustering algorithm with different neighborhood lists.

4.5. Best Results Comparison

Table 4 shows the best energy saving factor and coverage ratio loss results that can be achieved by the distinct mechanisms for our trace-driven test. Each mechanism uses distinct algorithms with a set of parameters as we have seen in the previous sections. As seen in Table 4, eSCIFI using cSCIFI+ achieved the highest energy saving factor among all mechanisms (64.32%) while presenting 0% coverage ratio loss.

ECMA, SEAR and eSCIFI using cSCIFI also achieved 0% coverage ratio loss. However, those mechanisms achieved lower energy saving factors when compared to the eSCIFI mechanism using cSCIFI+. SEAR got similar results to cSCIFI+ and better results than eSCIFI using cSCIFI. However, as explained previously and shown in Figure 9, SEAR energy saving result is affected by the neighborhood list ordering, while eSCIFI proposals using cSCIFI or cSCIFI+ are not affected at all. Therefore, our results show that eSCIFI using the cSCIFI+ algorithm achieved the best energy saving and coverage ratio loss results for our scenario.

Table 4. Mechanism's best results comparison.

Metrics	Baseline	ECMA	ACE	SEAR	cSCIFI	cSCIFI+
Coverage Ratio Loss (CR %)	0.42	0	20.73	0	0	0
Energy Saving Factor (\overline{ESF})	6.98	17.72	43.27	60.75	53.60	64.32

5. Conclusions

We presented the eSCIFI energy saving mechanism and its main architecture. eSCIFI uses traffic demand estimations given by machine learning models to manage the energy state of APs and it was designed to cope with a broader variety of wireless networks, specially those that cannot collect traffic data in a real time manner and/or have a limited CPU power.

We evaluated the normalized energy saving factor and the coverage ratio loss of our proposed mechanism. We also reproduced and compared eSCIFI results to the ones achieved by ACE, ECMA and SEAR. Those results showed that, for the UFF SCIFI network scenario, eSCIFI produced the best results. The best energy saving mechanism was the eSCIFI using the cSCIFI+ mechanism that can save up to 64.32% of the total energy consumed in a week without affecting the network coverage and user's association capacity.

eSCIFI has not been tested and implemented in real network scenarios yet. As future work, we plan to implement eSCIFI on the UFF SCIFI controller and do some future experiments using the real network infrastructure. The practical usage will give us some real insights about how to properly tune eSCIFI parameters according to a real implementation.

We also plan to use more sophisticated metrics as the average throughput and delay to evaluate the network performance and user coverage on those real network tests. We hope that those tests and new features will allow us to fully understand the eSCIFI possibilities and overcome its limitations.

Author Contributions: Conceptualization, G.H.A. and D.C.M.-S.; methodology, G.H.A., F.B. and D.C.M.-S.; validation, G.H.A., F.B., D.C.M.-S. and L.C.S.M.; formal analysis, G.H.A.; investigation, G.H.A.; resources, G.H.A. and L.C.S.M.; data curation, G.H.A., F.B. and L.C.S.M.; writing—original draft preparation, G.H.A. and D.C.M.-S.; writing—review and editing, G.H.A., F.B., D.C.M.-S. and L.C.S.M.; supervision, F.B., D.C.M.-S. and L.C.S.M.; project administration, D.C.M.-S.; funding acquisition, D.C.M.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by the Research Foundation of the State of Rio de Janeiro (FAPERJ), the Research Foundation of the State of São Paulo (FAPESP), the Coordination for the Improvement of Higher Education Personnel (CAPES), CAPES PRINT, and the Brazilian National Council for Scientific and Technological Development (CNPq).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The UFF SCIFI network association datasets (for classification and regression) are available at <https://github.com/midiacom/UFF-SCIFI-Datasets>, accessed on 3 December 2021.

Acknowledgments: We thank FAPERJ, FAPESP, CAPES, CAPES PRINT, and CNPq for their financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Debele, F.G.; Li, N.; Meo, M.; Ricca, M.; Zhang, Y. Experimenting resource-demand strategies for green WLANs. *ACM SIGMETRICS Perform. Eval. Rev.* **2014**, *42*, 61–66. [CrossRef]
2. Jardosh, A.P.; Papagiannaki, K.; Belding, E.M.; Almeroth, K.C.; Iannaccone, G.; Vinnakota, B. Green WLANs: on-demand WLAN infrastructures. *Mob. Netw. Appl.* **2009**, *14*, 798–814. [CrossRef]
3. Lyu, F.; Fang, L.; Xue, G.; Xue, H.; Li, M. Large-scale full WiFi coverage: Deployment and management strategy based on user spatio-temporal association analytics. *IEEE Internet Things J.* **2019**, *6*, 9386–9398. [CrossRef]
4. Lorincz, J.; Capone, A.; Bogarelli, M. Energy savings in wireless access networks through optimized network management. In Proceedings of the IEEE 5th International Symposium on Wireless Pervasive Computing, Modena, Italy, 5–7 May 2010; pp. 449–454.
5. Cui, Y.; Ma, X.; Wang, H.; Stojmenovic, I.; Liu, J. A survey of energy efficient wireless transmission and modeling in mobile cloud computing. *Mob. Netw. Appl.* **2013**, *18*, 148–155. [CrossRef]
6. Jardosh, A.P.; Iannaccone, G.; Papagiannaki, K.; Vinnakota, B. Towards an energy-star WLAN infrastructure. In Proceedings of the Eighth IEEE Workshop on Mobile Computing Systems and Applications, Tucson, AZ, USA, 8–9 March 2007; pp. 85–90.
7. Gomez, K.; Sengul, C.; Bayer, N.; Riggio, R.; Rasheed, T.; Miorandi, D. MORFEO: Saving energy in wireless access infrastructures. In Proceedings of the 2013 IEEE 14th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Madrid, Spain, 4–7 June 2013; pp. 1–6.
8. Dalmasso, M.; Meo, M.; Renga, D. Radio resource management for improving energy self-sufficiency of green mobile networks. *ACM SIGMETRICS Perform. Eval. Rev.* **2016**, *44*, 82–87. [CrossRef]
9. Rossi, C.; Casetti, C.; Chiasserini, C.F.; Borgiattino, C. Cooperative energy-efficient management of federated WiFi networks. *IEEE Trans. Mob. Comput.* **2015**, *14*, 2201–2215. [CrossRef]
10. Chen, Y.J.; Shen, Y.H.; Wang, L.C. Achieving energy saving with QoS guarantee for WLAN using SDN. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 23–27 May 2016; pp. 1–7.
11. Chin, K.W. A green scheduler for enterprise WLANs. In Proceedings of the 2011 IEEE Australasian Telecommunication Networks and Applications Conference (ATNAC), Melbourne, VIC, Australia, 9–11 November 2011; pp. 1–3.
12. Fang, L.; Xue, G.; Lyu, F.; Sheng, H.; Zou, F.; Li, M. Intelligent Large-Scale AP Control with Remarkable Energy Saving in Campus WiFi System. In Proceedings of the 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), Singapore, 11–13 December 2018; pp. 69–76.
13. Lee, K.; Kim, Y.; Kim, S.; Shin, J.; Shin, S.; Chong, S. Just-in-time WLANs: On-demand interference-managed WLAN infrastructures. In Proceedings of the IEEE INFOCOM 2016—the 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–14 April 2016; pp. 1–9.

14. Silva, P.; Almeida, N.T.; Campos, R. Energy consumption management for dense Wi-Fi networks. In Proceedings of the 2019 Wireless Days (WD), Manchester, UK, 24–26 April 2019; pp. 1–8.
15. Xu, C.; Han, Z.; Zhao, G.; Yu, S. A sleeping and offloading optimization scheme for energy-efficient WLANs. *IEEE Commun. Lett.* **2016**, *21*, 877–880. [[CrossRef](#)]
16. Xu, C.; Wang, J.; Zhu, Z.; Niyato, D. Energy-Efficient WLANs with resource and re-association scheduling optimization. *IEEE Trans. Netw. Serv. Manag.* **2019**, *16*, 563–577. [[CrossRef](#)]
17. Magalhães, L.C.S.; Balbi, H.D.; Corrêa, C.; Valle, R.D.T.d.; Stanton, M. Scifi—A software-based controller for efficient wireless networks. In *UbuntuNet-Connect 2013*; UbuntuNet Alliance: Kigali, Rwanda, 2013.
18. Fainelli, F. The OpenWrt embedded development framework. In Proceedings of the Free and Open Source Software Developers European Meeting, Brussels, Belgium, 23–24 February 2008; p. 106.
19. Apostolo, G.H.; Bernardini, F.; Magalhães, L.C.S.; Muchaluat-Saade, D.C. An Experimental Analysis for Detecting Wi-Fi Network Associations Using Multi-label Learning. In Proceedings of the IEEE 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Niterói, Brazil, 1–3 July 2020; pp. 423–428.
20. Wang, W.; Chen, J.; Hong, T.; Zhu, N. Occupancy prediction through Markov based feedback recurrent neural network (M-FRNN) algorithm with WiFi probe technology. *Build. Environ.* **2018**, *138*, 160–170. [[CrossRef](#)]
21. Zou, H.; Jiang, H.; Yang, J.; Xie, L.; Spanos, C. Non-intrusive occupancy sensing in commercial buildings. *Energy Build.* **2017**, *154*, 633–643. [[CrossRef](#)]
22. Ghai, S.K.; Thanayankizil, L.V.; Seetharam, D.P.; Chakraborty, D. Occupancy detection in commercial buildings using opportunistic context sources. In Proceedings of the 2012 IEEE International Conference on Pervasive Computing and Communications Workshops, Lugano, Switzerland, 19–23 March 2012; pp. 463–466.
23. Balaji, B.; Xu, J.; Nwokafor, A.; Gupta, R.; Agarwal, Y. Sentinel: occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings. In Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems, Roma, Italy, 11–15 November 2013; pp. 1–14.
24. Donevski, I.; Vallero, G.; Marsan, M.A. Neural networks for cellular base station switching. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Paris, France, 29 April–2 May 2019; pp. 738–743.
25. Budzisz, Ł.; Ganji, F.; Rizzo, G.; Marsan, M.A.; Meo, M.; Zhang, Y.; Koutitas, G.; Tassiulas, L.; Lambert, S.; Lannoo, B.; others. Dynamic resource provisioning for energy efficiency in wireless access networks: A survey and an outlook. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 2259–2285. [[CrossRef](#)]
26. Lorincz, J.; Bogarelli, M.; Capone, A.; Begušić, D. Heuristic approach for optimized energy savings in wireless access networks. In *SoftCOM 2010, Proceedings of the 18th International Conference on Software, Telecommunications and Computer Networks, Split-Bol, Croatia, 23–25 September 2010*; IEEE: New York, NY, USA, 2010; pp. 60–65.
27. Capone, A.; Malandra, F.; Sansò, B. Energy savings in wireless mesh networks in a time-variable context. *Mob. Netw. Appl.* **2012**, *17*, 298–311. [[CrossRef](#)]
28. Liu, L.; Xu, S.; Hu, J.; Cui, L.; Min, G. Balancing of the quality-of-service, energy and revenue of base stations in wireless networks via tullock contests. In Proceedings of the International Symposium on Quality of Service, Phoenix, AZ, USA, 24–25 June 2019; pp. 1–8.
29. Vallero, G.; Renga, D.; Meo, M.; Marsan, M.A. Greener RAN operation through machine learning. *IEEE Trans. Netw. Serv. Manag.* **2019**, *16*, 896–908. [[CrossRef](#)]
30. Tanaka, T.; Abe, K.; Aust, S.; Ito, T.; Yomo, H.; Sakata, S. Automatic and cooperative sleep control strategies for power-saving in radio-on-demand WLANs. In Proceedings of the 2013 IEEE Green Technologies Conference (GreenTech), Denver, CO, USA, 4–5 April 2013; pp. 293–300.
31. Yomo, H.; Kondo, Y.; Namba, K.; Tang, S.; Kimura, T.; Ito, T. Wake-up ID and protocol design for radio-on-demand wireless LAN. In Proceedings of the 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications-(PIMRC), Sydney, Australia, 9–12 September 2012; pp. 419–424.
32. Marsan, M.A.; Chiaraviglio, L.; Ciullo, D.; Meo, M. A simple analytical model for the energy-efficient activation of access points in dense WLANs. In Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking, Athens, Greece, 14–15 October 2010; pp. 159–168.
33. Kumazoe, K.; Nobayashi, D.; Fukuda, Y.; Ikenaga, T.; Abe, K. Multiple station aggregation procedure for radio-on-demand WLANs. In Proceedings of the IEEE 2012 Seventh International Conference on Broadband, Wireless Computing, Communication and Applications, Victoria, BC, Canada, 12–14 November 2012; pp. 156–161.
34. Kumazoe, K.; Nobayashi, D.; Fukuda, Y.; Ikenaga, T. Station aggregation scheme considering channel interference for radio on demand networks. In Proceedings of the 2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), Victoria, BC, Canada, 27–29 August 2013; pp. 265–270.
35. Ganji, F.; Budzisz, Ł.; Wolisz, A. Assessment of the power saving potential in dense enterprise WLANs. In Proceedings of the IEEE 24th International Symposium on Personal, Indoor, and Mobile Radio Communications, London, UK, 8–11 September 2013.
36. Nagareda, R.; Hasegawa, A.; Shibata, T.; Obana, S. A proposal of power saving scheme for wireless access networks with access point sharing. In Proceedings of the 2012 IEEE International Conference on Computing, Networking and Communications (ICNC), Maui, HI, USA, 30 January–2 February 2012; pp. 1128–1132.

37. Zhang, Y.; Jiang, C.; Wang, J.; Han, Z.; Yuan, J.; Cao, J. Green Wi-Fi implementation and management in dense autonomous environments for Smart Cities. *IEEE Trans. Ind. Inform.* **2017**, *14*, 1552–1563. [[CrossRef](#)]
38. Wu, W.; Luo, J.; Dong, K.; Yang, M.; Ling, Z. Energy-efficient user association with congestion avoidance and migration constraint in green WLANs. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 9596141. [[CrossRef](#)]
39. Apostolo, G.H.; Bernardini, F.; Magalhães, L.C.S.; Muchaluat-Saade, D.C. A Unified Methodology to Predict Wi-Fi Network Usage in Smart Buildings. *IEEE Access* **2021**, *9*, 11455–11469. [[CrossRef](#)]
40. Haratcherev, I.; Fiorito, M.; Balageas, C. Low-power sleep mode and out-of-band wake-up for indoor access points. In Proceedings of the 2009 IEEE Globecom Workshops, Honolulu, HI, USA, 30 November–4 December 2009; pp. 1–6.