Flow-Based Management For Energy Efficient Campus Networks

Ahmed Amokrane, Rami Langar, Raouf Boutaba, and Guy Pujolle

Abstract—Recent studies have shown that the energy consumption of wireless access networks is a threat to the sustainability of mobile cloud services. Consequently, energy efficient solutions are becoming crucial for both local and wireless access networks. In this paper, we propose a flow-based management framework to achieve energy efficiency in campus networks. We address the problem from the dynamic perspective, where users come and leave the system in an unpredictable way. Specifically, we propose an online flow-based routing approach that allows dynamic reconfiguration of existing flows as well as dynamic link rate adaptation, while taking into account users' demands and mobility. Our approach is compliant with the emerging software defined networking (SDN) paradigm since it can be integrated as an application on top of an SDN controller. To achieve this, we first formulate the flow-based routing problem as an integer linear program (ILP). As this problem is known to be $\mathcal{NP}$-hard, we then propose a simple yet efficient ant colony-based approach to solve the formulated ILP. Through extensive simulations, we show that our proposed approach is able to achieve significant gains in terms of energy consumption, compared to heuristic solutions and conventional routing solutions such as the shortest path (SP) routing, the minimum link residual capacity routing metric (MRC), and the load balancing (LB) scheme. In particular, we show that the energy consumption can be reduced by up to 7%, 35%, 44%, and 49% compared to Greedy-OFER, MRC, SP, and LB, respectively, while ensuring the required quality of service (QoS).

Index Terms—Energy efficiency, campus networks, management, SDN, optimization.

I. INTRODUCTION

THE UNPRECEDENTED expansion of broadband communication networks has led to a significant increase in energy consumption of communication networks. Indeed, according to a recent report [1], the Information and Communication Technologies (ICT) ecosystem is approaching 10% of the world's electricity usage. This corresponds to the same amount of power used to light the planet in 1985 and over 50% of the aviation consumption nowadays. Moreover,

based on current trends, wireless access technologies such as WiFi will soon be the dominant methods for accessing emerging cloud services [2]. In fact, according to a study published in 2013 [2], wireless cloud services energy consumption, in which wireless access networks represent 90%, will increase by 460% in 2015 to reach 43 TWh, up from 9.2 TWh only in 2012. To make things worse, this will result in an increase in carbon footprint from 6 megatonnes of CO_2 in 2012 to up to 30 megatonnes of CO_2 in 2015, which represents the equivalent of adding 4.9 million cars on the roads. Facing the fact that the cost of energy continues to rise and the increasing environmental awareness, operators and institutions are urged to reduce the energy consumption of their campus networks including enterprise campuses, school campuses, shopping malls, airports, etc [3], to reduce operational expenditures (OPEX) and achieve long term sustainability.

The application of green and energy efficient networking to campus networks has seldom been reported in the literature. A topology of a typical campus network is illustrated in Fig. 1. Typically, it comprises static Access Points (APs), a set of switches, and gateway routers. Each AP serves multiple mobile users and connects them directly or through a multi-hop wireless routing to the wired backbone. The wired backbone itself is composed of a set of switches that form more or less a hierarchy and converge towards gateway routers. The gateway routers ensure the forwarding of the traffic towards the Internet. It is worth noting that this topology can be used in enterprise or university campuses to enable cost-effective and scalable deployment of secure outdoor wireless LANs, as highlighted by Cisco [4]. For example, Concordia University (Canada) used a wireless mesh network in its campus [5].

According to recent studies [2], [6], user traffic drained by campus networks is expected to soar in the next few years, which will result in high energy consumption [2], [7]. As such, it is important to design energy efficient planning and management strategies for campus networks that take into account the dynamic and unpredictable users' mobility.

On the other hand, Software Defined Networking (SDN) [8] has emerged recently as a solution facilitating network management. The key idea behind SDN is to move the forwarding intelligence into a centralized network controller, while keeping the routers or switches simple. This allows to implement different forwarding approaches flexibly and achieve global optimizations easily. In SDN, the controller dictates the forwarding rules of flows to the forwarding elements using protocols such as OpenFlow [9]. SDN presents an opportunity to improve the performance and reduce the energy consumption of campus networks [10].
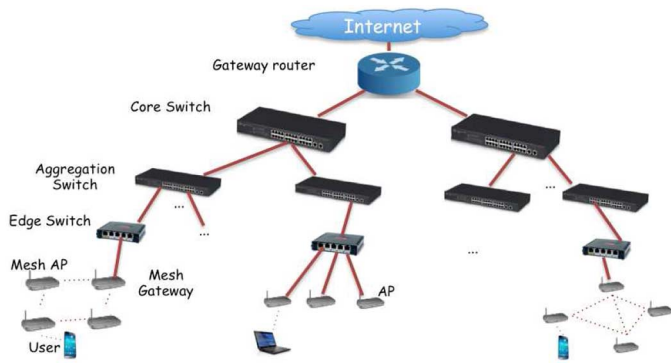
Fig. 1. Typical campus network topology.

Motivated by the potential of the new SDN paradigm, we propose in this paper a holistic energy conservation approach that uses online flow-based routing and link rate adaptation in campus networks. Our objective is to minimize the energy consumption of APs and switches, while routing incoming flows subject to QoS constraints (such as bandwidth and delay) and taking into account the dynamic and unpredictable arrival, departure and users' mobility. More specifically, our approach determines the AP to which each user will associate, along with a complete path in the wireless and wired parts of the network, towards the Internet, while minimizing the whole energy consumption and satisfying the QoS constraints. In this case, our proposed solution uses dynamic flow consolidation to reduce the energy consumption in the network. Moreover, it uses link rate adaptation to further reduce the energy consumption in the wired part of the network. Our proposed approach can be easily integrated in an SDN solution since it relies on a central controller that monitors and manages the network and decides on flow routes and link rates.

To achieve this, we first formulate the problem as an integer linear program (ILP), whose objective is to reduce the total energy consumption in the wireless and wired parts of the network. Moreover, The formulated objective function takes into account the costs for switching between sleeping and active modes of nodes (APs, switches and gateway routers), as well as re-routing or consolidating existing flows. As this problem is known to be $\mathbb{NP}$-hard [11], [12], we then propose a simple yet efficient algorithm based on Ant Colony, called Ant Colony Online Flow-based Energy efficient Routing (AC-OFER) to solve the formulated ILP problem. In this context, Greedy-OFER, Shortest Path (SP) routing strategy, the Minimum link Residual Capacity routing metric (MRC) and the Load Balancing (LB) schemes are used to develop baselines to which the AC-OFER improvements are compared. Through extensive simulations, we show that our proposed approach can achieve significant gains in terms of energy consumption. Specifically, the gains, can attain 6.5%, 17%, 42% and 45% compared to Greedy-OFER, MRC, SP and LB, respectively, for medium-sized networks. These gains become 7%, 35%, 44% and 49%, respectively, in large-sized networks, while achieving the same users' requests acceptance ratio and QoS satisfaction.

In summary, our key contributions are the following:

- We formulate the problem of reducing energy consumption in campus networks as an Integer Linear Program (ILP), under dynamic arrival and departure of users. The objective includes the cost of re-routing or consolidating existing flows as well as the cost of switching nodes between active and sleep modes.
- We propose a meta-heuristic low time complexity approach based on Ant Colony to solve the ILP problem. We show that our approach achieves near optimal solution for network reconfiguration to reduce the overall energy consumption within few milliseconds, which makes it usable in practice.
- Our approach is online flow-based and defines users attachment as well as flows routing in both wireless and wired parts of the network under dynamic arrival and departure of users, and uses link rate adaptation in the wired backbone to achieve energy efficient campus networks.
- We compare our approach with several existing solutions and discuss the associated gains.

The reminder of this paper is organized as follows. Section II presents an overview of the related work. In Section III, we describe the system model used in our analysis. Section IV formulates the problem as an ILP, followed by a presentation of our proposal in Section V. Simulation results are presented in Section VI. Finally, Section VII concludes this paper.

II. RELATED WORK

Energy management has been an active research area in the last few years. In the following, we survey relevant research in both wired and wireless networks.

A. Energy Reduction in Wired Networks

Numerous proposals have been presented to reduce energy consumption in wired networks [13]–[19]. Authors in [13] propose to reduce energy consumption in backbone networks by reducing the number of used nodes. They formulate the problem as an ILP for multi commodity flow and provide the optimal routing to reduce the number of used nodes. Authors in [14] propose to shut down nodes one by one and verify that the network still route the required traffic (i.e., the constraints are not violated). In [15], authors investigated a model based on gradient optimization to reduce energy consumption in wired networks. They started from routing paths given by a shortest path routing. Then, they used a routing policy named Energy-Aware Routing Protocol (EARP) [16] to reduce energy consumption by up to 10% given that the required QoS is satisfied. Online flow-based routing has also been used in [17], where authors presented E^2-MCRA, a flow-based routing approach that reduces the number of used nodes in a Internet Service Provider (ISP) network, while satisfying the QoS constraints. The idea is to route incoming flows by choosing among the possible paths, the one that achieves the best combination between the path length and the number of additional nodes to turn on. Authors in [18] proposed an ILP formulation to reduce the number of used nodes in the network while

routing per-flow basis. They also derived heuristics for online routing of flows. They used results from real routers to assess their proposed approach. Authors in [19] addressed the problem of energy reduction in ISP networks. Their proposal is time-driven and relies on the observation that ISP networks exhibit regular/predictive traffic patterns during specified time windows. Hence, they first propose a heuristic that shuts down unnecessary links, and then an algorithm to compute the duration of the time window. Then, they show that their approach can achieve up to 18% energy saving (in terms of used links) without significantly impacting the network performance.

On the other hand, energy efficiency in data center networks has also been addressed by a few works. The objective is to reduce the number of used nodes (both servers and switches) through virtualization and flow consolidation. Specifically, authors in [20], presented ElasticTree, a network-wide power manager that relies on OpenFlow to dynamically adjust the set of active network elements (i.e., links and switches) to satisfy changing data center traffic loads. They show that ElasticTree can save up to 50% of network energy, while maintaining the ability to handle traffic surges. In [21], authors propose a two step solution to reduce energy consumption in datacenters. First, they propose to assign virtual machines (VMs) to servers to reduce the amount of traffic and to generate favorable conditions for traffic engineering. Then, to achieve energy conservation, they reduce the number of needed active switches that balance traffic flows, depending on the relationship between power consumption and routing. In [22], authors showed the benefit of using link rate adaptation in a datacenter network. The objective is to find the rates of the different links that minimize the energy consumption of the whole network. They formulate the problem as an ILP and proposed an approximation method to solve the ILP. However, the proposed energy consumption model assumes proportional switch energy consumption (i.e., the energy consumption of the switch is proportional to the link rates), which is not true in practice.

In the specific case of campus networks, authors in [23] proposed an SDN-based approach (i.e., using a central controller) to reduce the energy consumption by switching off as many switches/routers as possible. They formulated the problem as a mixed integer linear programming (MILP) and then proposed a heuristic solution. However, authors did not address the case of wireless parts of the network. Moreover, no link rate adaptation is considered.

B. Energy Reduction in Wireless Networks

An important body of work on energy efficiency of devices and protocols in cellular, WLANs and Wireless Mesh Networks (WMNs) has been reported in the literature. A survey on energy efficient protocols for such networks can be found in [24]. In WLANs, authors in [25], [26] presented strategies based on the resource on-demand concept. The proposed approaches derive strategies to reduce the number of used APs in a WLAN. In [27], authors proposed an analytical model to assess the effectiveness of this concept and authors in [28] showed management strategies for energy savings in solar powered 802.11 WMNs. In [29], authors derived mobility and traffic patterns in the specific case of WLAN university campus networks (day/night, week days and weekends) to decide on the APs to turn on/off in the campus to accommodate the traffic. In cellular access networks, authors in [30] summarized existing energy saving approaches, which use carrier aggregation, turn off transmission components during signal-free symbols, and turn off cells during low traffic periods.

In the context of WMNs, relevant works on energy-efficiency are reported in [31]–[36]. These works consider offline routing in such networks. Specifically, authors in [31] consider the case of WMNs, where the users can choose the AP they connect to. To do so, they formulate and solve the problem as an ILP, where the objective is to minimize the number of used nodes (i.e., APs and gateways), and always satisfy users' bandwidth demands. However, they do not take into account the interference between APs since directional antennas are assumed. The authors extended this work in [32] to consider the planning and deployment of APs (i.e., choosing the number of APs and their location). Another energy management study in WMNs is provided in [33], where a combination between different modulation techniques and power adaptation is presented. In [34], we proposed a framework for energy efficient management in TDMA-based WMNs. However, all the aforementioned approaches are still limited since they are offline. Indeed, the traffic patterns are assumed to be known a priori and fixed at the planning stage, which is not usually the case in practice, as users can arrive and leave the network in an unpredictable way. This may limit the utilization of such approaches in real world deployments. Moreover, these approaches do not account for the wired part of a campus network.

To overcome these limitations, we first proposed in [35] a flow-based management framework to achieve energy efficient WMNs, that takes into account the dynamic and unpredictable traffic patterns. In this paper, we rather focus on campus networks, which include both wireless and wired networks elements. It is worth noting that a more detailed survey of energy efficient approaches in wireless access networks, with a focus more on cellular and WLAN, is given in [37].

C. SDN-Based Wireless Network Management

In the context of SDN, most of the proposed solutions to wireless network management focus on mobility management. For instance, authors in [38] present an OpenFlow-based approach for efficient mobility management in WMNs. This approach implements the mobility management as an application on top of the SDN controller. In [39], authors present an SDN-based framework for network management in WLANs. The framework relies on a central controller that has a global view of the network. The mobility management is implemented as an application on top of the central controller that manages users' attachments through light virtual APs and OpenFlow-enabled switches.

Adopting the same SDN paradigm as in [39] and [38], in this paper, we propose an online flow-based routing approach in campus networks. Specifically, we focus on energy efficient communications by routing the incoming traffic from the users to the campus gateway routers, while considering the required

QoS, the energy consumption as well as the costs of flows reconfiguration and re-routing. To the best of our knowledge, we are the first to address green routing in SDN-based campus networks.

III. SYSTEM MODEL

A. Network Model

We represent a campus network by a directed graph $G(V \cup W, E_s \cup E_d)$, called a connectivity graph, where V is the set of APs, W is the set of switches, E_s and E_d are the set of wireless and wired links, respectively. We denote also by $E = E_s \cup E_d$ the set of all links. Each node $v \in V$ represents an AP. Note that some of these APs can be interconnected to form a Wireless Mesh Network (WMN). A wireless link $e \in E_s$ between two APs has a number of channels denoted by nc_e. The capacity along each channel is limited and denoted by C_{ek}. Moreover, each AP $i \in V$ has a limited capacity to serve its attached users denoted by C_i.

Similarly, each wired link $e \in E_d$ between an AP and a switch or between two switches has a limited capacity denoted by C_{ek}. In this case, $nc_e = 1, \forall e \in E_d$. Moreover, some switches have ports connected to gateway routers, which guarantee the connection to the Internet. Let S_d denote this set of switches. One can think of these switches as the gateways towards the Internet for the wired backbone part of the network. Note that each gateway $j \in S_d$ router has a limited capacity for traffic forwarding towards the Internet denoted by G_j.

B. AP Energy Consumption Model

First of all, recall that an AP $v \in V$ has two physical interfaces: one for serving its mesh users (called AP interface) and one for relaying traffic in the wireless backbone towards the wireless backbone (called mesh interface). An additional third wired interface exists for APs that are directly connected to the switches to forward traffic to/from the wired network.

Given an AP $v \in V$, we distinguish between two operating modes: low power consumption and high power consumption. In the first mode, an AP has no users attached to it and no traffic to forward. In this case, it only uses its AP Interface to detect user's presence. In this mode, the energy consumption of the AP can be reduced by setting up a high sleeping period and reducing the transmission power as presented in [40], or shut down the AP by cutting down the power supply such as the Power over Ethernet (PoE). In the second mode, the AP has either active users attached to it or traffic to forward. Therefore, its power consumption is higher. It is worth noting that the contrast between low power and high power consumption stems from the study carried out by Gomez et al. [41]. Through real power measurements, authors showed that the power consumption of an AP when it does not carry traffic (i.e., active AP with traffic load equals to 0), represents 75% of its peak power. An active AP carrying traffic consumes an extra power proportional to the AP traffic load, and is calibrated as 25% of the power consumption multiplied by the traffic load over total AP capacity, as shown in the revised paper.

TABLE I
AP POWER CONSUMPTION PROFILES

Profile	Explanation
P_R	If v is used as a mesh router only. This means that v has traffic to forward in the wireless backbone but does not have users attached to it.
P_{AR}	If v is used as an access point and a mesh router at the same time. This means that v has active users attached to it and uses its mesh interface to forward traffic in the wireless backbone.
P_{AG}	If v is used as an access point and has traffic to forward to the wired network at the same time (i.e., the AP is acting as a mesh gateway and has active users attached to it).
P_{RG}	If v is used as a mesh router and mesh gateway at the same time but no users are attached to it.
P_{ARG}	If v is used as an access point, a mesh router and a mesh gateway at the same time. In this case, v uses simultaneously its three interfaces. This is the most power hungry profile.
P_S	If v has no active users attached to it, no traffic to forward and is not used as a mesh gateway. This is the power saving mode of an AP. Note that P_S could be negligible compared to the other profiles.

Hence, six power consumption profiles for an AP $v \in V$ can be defined, and listed in Table I. Note that we denote by mesh router an AP that has traffic to forward in the wireless backbone and a mesh gateway if it has traffic to forward from the wireless backbone towards the wired network.

Consequently, the power consumption of AP can be expressed as follows:

$$P = \begin{cases} P_S & \text{If } load_{AP} = 0 \\ P_{profile} * (0.75 + 0.25 \times \frac{load_{AP}}{total\ AP\ capacity}), & \text{Otherwise} \end{cases}$$

where $load_{AP}$ is the current load of the AP, *total AP capacity* is the maximum capacity supported by the AP and $P_{profile}$ is the peak power consumption of the corresponding profile.

It is worth noting that in the case of one-hop wireless networks (i.e., all APs are directly connected to switches such as the topology discussed in [10]), the energy consumption modes are reduced to P_{AG} and P_S.

To reduce the energy consumption of the whole network, one should put as many nodes as possible into power saving mode and by switching them off in case where they have no traffic to carry. In practice, this can be achieved using technologies such as Wake-on-LAN or Power over Ethernet (PoE) powered APs as in [3], [7], which are switched off by cutting the power Ethernet supply.

C. Switch Energy Consumption Model

A switch contains a number of interfaces. Each interface has its own Network Interface Card (NIC) card (a.k.a. line card). Each interface can have one or multiple ports. In this work, we use the energy consumption model proposed by Mahadevan et al in [42] and reused in other works such as [43]. In this model, the energy consumption of a switch corresponds to a fixed amount of power consumed by the chassis, plus a variable amount of power that depends on the number of active interfaces along with the rate of each interface. It is given by:

$$P_{switch} = P_{chassis} + n_{linecards} \times P_{linecards} + \sum_{k=1}^{R} n_{ports.r} \times P_r$$

(1)

where $P_{chassis}$ is related to the power consumed by the switch hardware, $P_{linecards}$ is the power consumed by an active network line card, and P_r corresponds to the power consumed by a port (transceiver) running at rate $r \in \{r_1, \ldots, r_m\}$.

Note that in equation (1), only the last component appears to be dependent on the link rate, while other components, such as $P_{chassis}$ and $P_{linecards}$, remain fixed for the whole switch operation duration.

D. Traffic Model

In this work, we model the traffic as a set of L flows. Each flow originates from a user, who is located in the coverage area of one or multiple APs. Users' location is captured by the coverage matrix A. Each flow $l \in L$ has a bandwidth demand b_l and a delay constraint d_l. Note that these flows are unidirectional. As such, the downlink and uplink are considered to be two different flows and are treated differently in terms of allocated path. Moreover, we assume that the traffic demand of each flow can be determined by the controller. For instance, this could be achieved by using the counters per flow in the network forwarding elements, and use estimation techniques to determine the traffic rate.

IV. PROBLEM FORMULATION

As already mentioned, our objective is to minimize the energy consumption of the network nodes (i.e., APs and switches) over time, while routing dynamically the arriving and departing flows subject to QoS constraints (i.e., bandwidth and delay). In other words, our objective is to reduce the energy consumption, while guaranteeing traffic routing and QoS for the different traffic flows in campus networks. More specifically, the problem can be formulated as follows:

GIVEN:
- A physical topology represented by the graph $G(V \cup W, E_s \cup E_d)$, which is described by the connectivity and interference matrices M and I, respectively.
- A set of gateway routers in the wired backbone network.
- A set L of flows originating from users, each one with its bandwidth demand b_l and delay constraint d_l.
- The coverage matrix A of APs.
- The current attachment of users and their flows' routes.

FIND:
- The optimal attachment of each user to one of the APs and, optimal routing of its flows that minimizes the network operation and reconfiguration costs, subject to QoS constraints (i.e., bandwidth and delay), and the link rates of the wired link in the network.

In the following, we formulate the flow-based routing problem as an integer linear program (ILP). For ease of understanding, table II summarizes the symbols used in our analysis.

Let t be the epoch starting when one of the following events occurs: user arrival/departure or user movement between two APs. We denote by $t - 1$ the epoch before t. For the sake of presentation, let us use the notation y and y' to designate the state of any variable y at epoch t and $t - 1$, respectively.

Notation	Meaning
E_s	The set of wireless links
E_d	The set of wired links
$f_{e,k,l}$	0 or 1, whether the channel k of a link e is used to route flow l
w_{li}	0 or 1, whether the user originating the flow l is attached to the AP i
y_i	0 or 1, whether an AP is in the active or sleep mode
y_i^+	0 or 1, whether an AP is turned into the active mode
y_i^-	0 or 1, whether an AP is turned into the sleep mode
r_{il}^+	0 or 1, whether an AP i is added to the path of flow l
r_{il}^-	0 or 1, whether an AP i is removed from the path of flow l
nc_e	Number of channels of link e
C_{ek}	Capacity of channel k of link e
C_i	Capacity of AP i
G_j	Capacity of gateway j
r_e	The rate at which link $e \in E_d$ is set to operate

We introduce the binary variable w_{li} to indicate whether a user originating flow l is attached to the AP $i \in V$ as follows:

$$w_{li} = \begin{cases} 1 & \text{If user originating flow } l \text{ is attached to AP } i \\ 0 & \text{Otherwise} \end{cases}$$

To represent the link and channel allocation, we define another binary variable $f_{e,k,l}$, which takes the value of 1 whenever the flow l uses the channel k on link e on its route.

$$f_{e,k,l} = \begin{cases} 1 & \text{If flow } l \text{ is routed though link } e \text{ using channel } k \\ 0 & \text{Otherwise.} \end{cases}$$

Recall that $k = 1, \forall e \in E_d$. To define the link rate (i.e., the rate at which the link is set to operate) of a link $e \in E_d$, we use r_e. Recall that $r_e \in \{r_1, \ldots, r_m\}$.

To indicate whether an AP $i \in V$ is used or not, we introduce another binary variable y_i defined by:

$$y_i = \begin{cases} 0 & \text{If } \sum_{l \in L} \sum_{\substack{e \in E \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} + \sum_{l \in L} w_{li} = 0 \\ 1 & \text{Otherwise.} \end{cases}$$

where $E = E_s \cup E_d$, $s(e)$ and $d(e)$, respectively, denote the source and destination of link $e \in E$, and $i \in \{s(e), d(e)\}$ denotes that AP $i \in V$ is the source or destination of the link $e \in E$.

Let us consider the variable y_i^+ and y_i^- that represent, respectively, the decision of switching an AP i to active mode or sleep mode, at network reconfiguration. They are defined as follows:

$$y_i^+ = \begin{cases} 1 & \text{if } y_i' = 0 \text{ and } y_i = 1 \\ 0 & \text{Otherwise.} \end{cases}$$

$$y_i^- = \begin{cases} 1 & \text{if } y_i' = 1 \text{ and } y_i = 0 \\ 0 & \text{Otherwise.} \end{cases}$$

Note that switching a node from a sleep mode to active mode and vice versa generates a cost. This cost is denoted by cs_i^+ and cs_i^-, respectively, and could be the time needed to turn on the node or the energy that is consumed to set up the routing tables (e.g., flow table). In addition, we need to account for a cost when re-routing a flow over a more favorable route. Hence, let us consider the variables r_{il}^+ and r_{il}^- representing,

respectively, whether a flow l is re-routed through node i, after network reconfiguration, or removed from being routed through node i. They are defined as follows:

$$r_{il}^+ = \begin{cases} 1 & \text{If } \sum_{\substack{e \in E \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l}' = 0 \\ & \text{and } \sum_{\substack{e \in E \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \\ 0 & \text{Otherwise.} \end{cases}$$

$$r_{il}^- = \begin{cases} 1 & \text{If } \sum_{\substack{e \in E \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l}' \geq 1 \\ & \text{and } \sum_{\substack{e \in E \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} = 0 \\ 0 & \text{Otherwise.} \end{cases}$$

The re-routing costs will be thus represented by cr_{il}^+ and cr_{il}^-, respectively.

The power consumption of an AP $i \in V$ is given by P_i as follows:

$$P_i = \begin{cases} P_R & \text{If } \sum_{\substack{e \in E_s \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \text{ and } \sum_{l \in L} w_{li} = 0 \\ & \text{and } \sum_{\substack{e \in E_d \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} = 0 \\ P_{AR} & \text{If } \sum_{\substack{e \in E_s \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \text{ and } \sum_{l \in L} w_{li} \geq 1 \\ & \text{and } \sum_{\substack{e \in E_d \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} = 0 \\ P_{AG} & \text{If } \sum_{\substack{e \in E_s \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} = 0 \text{ and } \sum_{l \in L} w_{li} \geq 1 \\ & \text{and } \sum_{\substack{e \in E_d \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \\ P_{RG} & \text{If } \sum_{\substack{e \in E_s \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \text{ and } \sum_{l \in L} w_{li} = 0 \\ & \text{and } \sum_{\substack{e \in E_d \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \\ P_{ARG} & \text{If } \sum_{\substack{e \in E_s \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \text{ and } \sum_{l \in L} w_{li} \geq 1 \\ & \text{and } \sum_{\substack{e \in E_d \\ i \in \{s(e), d(e)\}}} \sum_{k=1}^{nc_e} f_{e,k,l} \geq 1 \\ 0 & \text{Otherwise.} \end{cases}$$

Recall that the power consumption of a switch $i \in W$ is given by the consumption model given in equation (1).

We now formulate the problem of routing the new incoming flow and dynamically re-optimizing the existing flows as an ILP with the following objective function:

$$Minimize \left(\alpha_E \sum_{i \in V \cup W} P_i + \alpha_S \sum_{i \in V \cup W} (y_i^+ cs_i^+ + y_i^- cs_i^-) \right.$$
$$\left. + \alpha_R \sum_{i \in V \cup W} \sum_{l \in L} (r_{il}^+ cr_{il}^+ + r_{il}^- cr_{il}^-) \right) \quad (2)$$

Where, α_E, α_S and α_R respectively represent weight factors to achieve a tradeoff between power consumption, re-routing flows and switching nodes to different states. For instance, they might represent the cost in USD (e.g., cost of electricity for α_E) or the cost translated into USD for any service disruption that might happen in the network if a flow is rerouted through a different path. Note that the first term in the objective function is related to the energy consumption when using a node i. The second term corresponds to the cost of switching nodes from sleeping/active states after reconfiguration, and the third term captures the cost of re-routing flows. Consequently, if a network administrator is interested only in the energy consumption and he is willing to give away possible service disruptions, he could set α_S and α_R to 0 and α_E to 1 and have an objective function based only on energy consumption.

The optimization is subject to the following constraints (3)–(13):

- Not exceeding the capacities of links and channels:

$$\sum_{l \in L} f_{e,k,l} \times b_l \leq C_{ek}, \ \forall e \in E_s \cup E_d, \forall k \in \{1, .., nc_e\} \quad (3)$$

- Not exceeding gateway routers capacities:

$$\sum_{l \in L} \sum_{e \in E} \sum_{k=1}^{nc_e} f_{e,k,l} \times b_l + \sum_{l \in L} w_{li} \times b_l \leq G_i$$
$$\forall e \in E_d, d(e) = i \text{ and } i \in S_d \quad (4)$$

- Not exceeding the APs capacities:

$$\sum_{l \in L} w_{li} \times b_l \leq C_i \quad \forall i \in V \quad (5)$$

- A user can attach to, at most, one AP that covers its location:

$$\sum_{i \in V} w_{li} \leq A_{li}, \quad \forall l \in L \quad (6)$$

- The delay constraint of a flow l should be satisfied:

$$\sum_{e \in E} \sum_{k=1}^{nc_e} f_{e,k,l} \leq d_l, \quad \forall l \in L \quad (7)$$

- A flow is not routed when it reaches a gateway unless the gateway capacity is exceeded, in which case the traffic is forwarded to another gateway:

$$\sum_{l \in L} \sum_{\substack{e \in E_d \\ d(e)=i}} \sum_{k=1}^{nc_e} f_{e,k,l} \times b_l + \sum_{l \in L} w_{li} \times b_l$$
$$\leq G_i + \sum_{l \in L} \sum_{\substack{e \in E_d \\ s(e)=i}} \sum_{k=1}^{nc_e} f_{e,k,l} \times b_l , \ \forall i \in S_d \quad (8)$$

- No loops when routing. This means that a flow comes in or goes out from a node at most once. Hence, we have:

$$\sum_{\substack{e\in E \\ s(e)=i}} \sum_{k=1}^{nc_e} f_{e,k,l} \le 1, \; \sum_{\substack{e\in E \\ d(e)=i}} \sum_{k=1}^{nc_e} f_{e,k,l} \le 1,$$
$$\forall i \in V \cup W, \forall l \in L \quad (9)$$

- Flow conservation constraint, which ensures that the network flow that enters a node plus the traffic originating from this node is equal to the outgoing traffic from this node. It can be written as follows:

$$\sum_{l\in L} \sum_{\substack{e\in E \\ s(e)=i}} \sum_{k=1}^{nc_e} f_{e,k,l} \times b_l = \sum_{l\in L} \sum_{\substack{e\in E \\ d(e)=i}} \sum_{k=1}^{nc_e} f_{e,k,l} \times b_l$$
$$+ \sum_{l\in L} w_{li} \times b_l, \quad \forall i \in (V \cup W) \backslash S_d \quad (10)$$

- In the wireless part, two links that interfere with each other cannot transmit at the same time. This means that the sum of their proportion of link usage should not exceed 1.

$$\sum_{l\in L} \frac{f_{e,k,l} \times b_l}{C_{ek}} + \sum_{l'\in L} \sum_{e'\in E_s} \frac{f_{e',l',k} \times b_{l'} \times I_{(e,k),(e',k)}}{C_{e'k}} \le 1$$
$$\forall e \in E_s, \forall k \in \{1, .., nc_e\} \quad (11)$$

- In the wired part, the link rates should be adjusted to the upper rate r_i that satisfies the used bandwidth in the corresponding links.

$$r_e = \min_{j\in\{1..m\}} r_j \ge \sum_{l\in L} f_{e,k,l} \times b_l, \forall e \in E_d \quad (12)$$

- The decision variables are binary

$$f_{e,k,l}, \; w_{li} \in \{0,1\}, \forall i \in V, \forall e \in E, \forall l \in L \quad (13)$$

In a nutshell, constraints (3)–(5) guarantee not violating the capacity constraints of the network nodes. This means not routing traffic beyond the capacity of the network nodes and links. Constraint (6) guarantees that a user can attach to only one AP at a time if his traffic flow is routed. Constraint (7) guarantees that the delay requirement specified by a flow in terms of number of hops is satisfied. Constraints (8)–(10) guarantee that traffic flows are always routed to terminate at a gateway subject to the gateways capacities and that each flow is routed in a contiguous way through the network. Constraint (11) guarantees that the interfering links in the wireless part are not used at their full capacity since they share the wireless transmission medium. Constraint (12) defines how the link rates are set in the wired part of the network. Finally, constraint (13) guarantees that the decision variable are binary ones.

In the following, we present our meta-heuristic approach, called AC-OFER, that solves the above ILP problem.

V. AC-OFER PROPOSAL

The formulated ILP problem presented in Section IV is assumed to be solved by the network controller each epoch t (i.e., each incoming flow). Indeed, the time scale of t should be short enough to capture the dynamic of arrival and departure of clients, as the new flows should be routed at their arrival. Clearly, such approach is not feasible in practice, since it generates high overhead due to the frequent updates of the flow tables. In addition, the above ILP problem is $\mathbb{NP}$-hard [11], [12].

To overcome these issues, we propose a two step approach. First, each incoming flow is injected in the network without incurring any changes on the already established routes of existing flows by computing an energy efficient path. This step is referred to as *"Network Event Handling"*. Then, a simple yet efficient meta-heuristic algorithm, called AC-OFER, is executed at each pre-defined time period T (and not at each flow arrival or departure). This step is called *"Dynamic Network Reconfiguration"*. The benefit of doing so is twofold: (i) to reduce the overhead due to rerouting existing flows and (ii) to decide on flow rerouting that optimizes the overall energy consumption of the network by taking into account any rerouting costs. In the following, we detail these two steps.

A. Network Event Handling

Upon detecting the "user arrival" event, one or multiple served APs start by sending the corresponding flow QoS requirements (i.e., bandwidth and delay constraints) to the network controller. Depending on the AP location, we can use either a virtual interface if it is a mesh node as proposed and validated in [38], where each physical wireless interface can be split into two virtual interfaces, or the usual secure channel as in OpenFlow-enabled switches if it is a switch [9]. Since no dynamic reconfiguration is performed at this level, the network controller chooses, among the possible paths, the one with the minimum score given by the objective function in (2), without incurring any changes to existing flow routes. To do so, we use a modified version of the Dijkstra's algorithm presented in Algorithm 1. The algorithm takes as input the graph of the network (APs and switches) along with the residual capacities of the APs and links. Going through Algorithm 1, we first modify the graph $G(V \cup W, E_s \cup E_d)$ by adding a new node v_l with an edge between v_l and all the APs that can cover the originating user's location, and which has enough residual capacity. Then, we use the same process as in Dijkstra's algorithm. Note that the objective function given in (2) is used as a distance function. Indeed, $diff\_power(u, v)$ in Algorithm 1 refers to the additional score of the objective function if the node v is added to the path that goes through u for flow l. At the end, the algorithm returns the path that terminates at a gateway, satisfies the delay of flow l and has the shortest distance from the source v_l. It is worth noting that the complexity of Algorithm 1 is in

---

Algorithm 1. New Arriving Flow Route Computation

1: **IN:** Campus Network ($G(V \cup W, E_s \cup E_d)$) with residual capacities in the links and APs, a new l flow to route
2: **OUT:** A route for the new flow without changing existing flows
3: - Extend $G(V \cup W, E_s \cup E_d)$ by adding a new node v_l
4: - Add a virtual edge between v_l and the APs that can cover the location of the user l
5: *Initialization:*
6: **for all** v in $V \cup W$ **do**
7:    $power[v] \leftarrow \infty$
8:    $visited[v] \leftarrow false$
9:    $previous[v] \leftarrow undefined$
10: **end for**
11: $dist[l] = 0$
12: $Q.Enqueue(l)$
13: *Compute the paths:*
14: **while** not $Q.Empty()$ **do**
15:    $u \leftarrow Q.getSmallestDist()$
16:    *// get the node with the smallest distance in Q*
17:    remove u from Q
18:    $u.visited = true$
19:    **for all** v in neighbors u **do**
20:      **if** (u, v) satisfies the bandwidth demand of d **then**
21:       $temp \leftarrow power[u] + diff\_power(u, v)$
22:       *//diff_power(u, v) gives the additional power if we add the node v to the path that goes through u for flow l*
23:       **if** $temp < power[v]$ **then**
24:        $power[v] \leftarrow temp$
25:        $previous[v] \leftarrow u$
26:        **if** $visited[v] = false$ **then**
27:         $Q.Enqueue(v)$
28:        **end if**
29:       **end if**
30:      **end if**
31:    **end for**
32: **end while**
33: Adapt the link rates using Algorithm 2
34: **Return** the path with the smallest score in *power* and that satisfies delay constant of l and terminates at a gateway

---

Algorithm 2. Discrete Link Rate Adaptation

1: **IN:** Campus Network ($G(V \cup W, E_s \cup E_d)$)
2: **OUT:** link rates for the different links
3: **for all** e in E_d **do**
4:    **if** $utilization(e) \leq r_{10}$ **then**
5:      $rate[e] \leftarrow r_{10}$
6:    **end if**
7:    **if** $r_{10} < utilization(e) \leq r_{100}$ **then**
8:      $rate[e] \leftarrow r_{100}$
9:    **end if**
10:    **if** $r_{100} < utilization(e) \leq r_{1000}$ **then**
11:      $rate[e] \leftarrow r_{1000}$
12:    **else**
13:      $rate[e] \leftarrow r_{10000}$
14:    **end if**
15: **end for**

B. Dynamic Network Reconfiguration Using Ant Colony Online Flow-Based Energy efficient Routing (AC-OFER)

As stated before, in order to optimize the overall energy consumption and resource utilization, the network controller needs to reconfigure the flow routes in the network taking into account the cost of re-routing or consolidating existing flows. This is performed at each predefined time period T. Note that T is a parameter that is specified by the network administrator, and can be in the order of minutes or hours. To this end, we propose to approximate the optimal solution of the above-mentioned ILP problem presented in Section IV using an Ant Colony-based approach [46], called AC-OFER.

AC-OFER operates as illustrated in the flowchart given in Fig. 2. First, a set of solution components (i.e., paths) needs to be determined for each flow coming from a user. Next, A_{max} artificial ants are launched and iteratively explore the search space until a predetermined number of iterations N_{max} is reached. During each iteration, each ant among A_{max} incrementally constructs the solution by adding in every step one solution component (i.e., a path for one user's flow) to the partial solution constructed so far. Note that the solution component to add among the candidates is chosen using a stochastic local decision policy. More specifically, the decision is based on *heuristic* information, denoted by η, and artificial *pheromone trails*, denoted by τ, which respectively quantify the desirability of a priori and a posteriori transition. Indeed, the *heuristic* represents the attractiveness of the move, indicating the a priori desirability of that move. On the other hand, the *pheromone trails* indicate how proficient it has been in the past (i.e., according to other ants experience) to add that solution component. Once an ant has built a solution, or while the solution is being built, the ant evaluates the partial solution and deposits pheromone trails on the components it used. This pheromone information will direct the search of the future ants.

More formally, our AC-OFER algorithm is described by the pseudo-code in Algorithm 3. The fundamental steps of AC-OFER are: 1) Formation of solution components, 2) Probabilistic selection of the candidate, 3) Selection of the
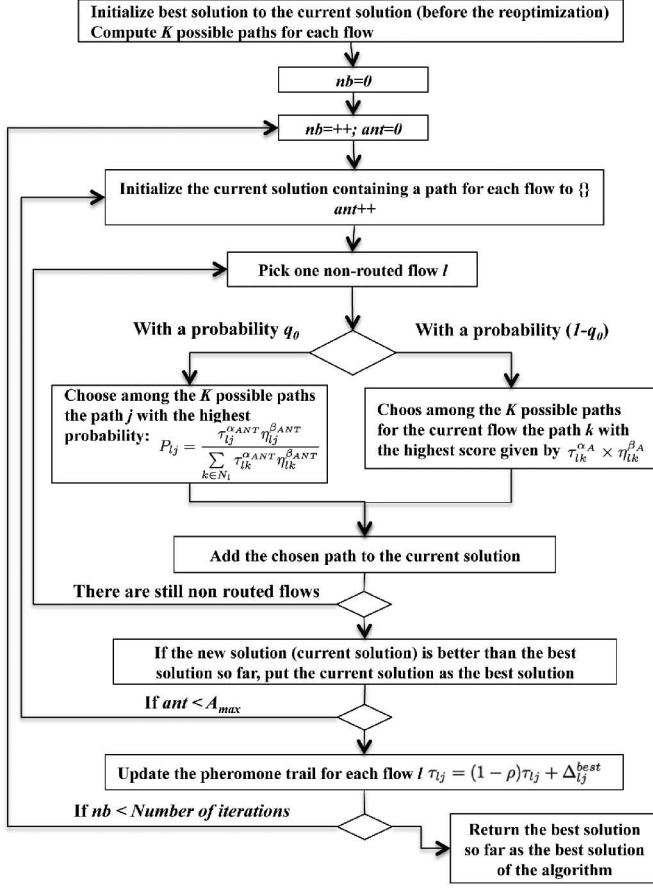
the order of $O((|V| + |W|)^2)$, where $|V|$ and $|W|$ are the total number of APs and switches in the network, respectively.

Once the path is chosen, the rates at the different links are adapted. In our study, and as proposed in previous works such as [44] and [45], we assume that the link rate (i.e., switch port) can be adjusted to one of the following predefined rates: 10 Mbps, 100 Mbps, 1 Gbps and 10 Gbps, denoted by $r_{10}, r_{100}, r_{1000}, r_{10000}$, respectively. Each one of these rates r_i represents one power consumption profile of the link (and thus of the corresponding switch ports) as used in equation (1). To do so, we propose an intuitive algorithm, presented in Algorithm 2, to set up link rates according to their utilization. Note that in case of user's departure, its corresponding flow will be removed from the network and the used resources will be released.

Fig. 2. AC-OFER flowchart diagram.

Algorithm 3. AC-OFER Algorithm

1: **IN:** Campus Network with routed flows (i.e., previous routes)
2: **OUT:** New routes solution (One path for each flow)
3: Set Parameters: q_0, α_A, β_A, Q
4: Initialize pheromone trails and best_solution to the previous solution
5: **for** $nb = 1 \rightarrow$ Number of Iterations **do**
6:     *//Construct Ant Solutions*
7:     **for all** ant in A_{max} **do**
8:         current_solution $\leftarrow \{\}$
9:         **for** $l = 1 \rightarrow$ Number of flows **do**
10:             $p \leftarrow Random(0..1)$
11:             **if** $p < q_0$ **then**
12:                 Choose path j among the K paths where
13:                 $j = \text{Argmax}_{k \in N_l}\left(\tau_{lk}^{\alpha_A} \times \eta_{lk}^{\beta_A}\right)$
14:             **else**
15:                 Choose path j according to P_{lj} given in (15)
16:             **end if**
17:             Add the jth path for flow l to current_solution
18:         **end for**
19:         **if** current_solution is better than best_solution **then**
20:             best_solution $\leftarrow$ current_solution
21:         **end if**
22:     **end for**
23:     *//Update Pheromones for all flows l*
24:     $\tau_{lj} \leftarrow (1 - \rho)\tau_{lj}$ *//Evaporate all pheromones*
25:     **if** current_solution is the best solution for the current iteration And j^{th} path is selected for flow l **then**
26:         $\tau_{lj} \leftarrow \tau_{lj} + \Delta_{lj}^{best}$
27:     **end if**
28: **end for**
29: **Return** best_solution

best solution and 4) Updating the pheromone trails. In the following, we detail these stages.

1) Formation of Solution Components: For each user, we consider K alternative paths towards a gateway (any of the m available gateways). Each path starts from the user, passes through an AP that the user attaches to, and then other intermediate APs then switches until reaching a gateway router. A solution component will be one of the predetermined K paths. As such, the number of possible solutions for the path formulation is $K^{|L|}$, where $|L|$ is the total number of lows, which is equal to the total number of users since each user is assumed to generate one flow. Hence the proposed meta-heuristic guides the algorithm to efficiently explore the graph of solutions.

2) Selection Among the Candidates for a Component: During each iteration, each ant among A_{max} builds the solution step by step, by adding in each step another component (i.e., a path for a flow l). The component to add is chosen according to the attractiveness of the new constructed solution (i.e., the current solution augmented by the selected component) which is called *the heuristic*, and the amount of *pheromone* deposits, which represents how this component is evaluated during the previous iterations by all ants. The *heuristic* is given by :

$$\eta = \frac{1}{\text{Objective Function Value}} \quad (14)$$

Once the objective function score computed, the choice of the next component to add to the partial solution constructed so far

(i.e., a path j for flow l) is selected according to a given probability. Note that in Ant Colony System meta-heuristic [46], two strategies can be used: *exploitation* and *exploration*. More specifically, *exploitation* is used with a probability q_0, whereas *exploration* is adopted with a probability $(1 - q_0)$.

Regarding *exploration*, the knowledge and experience of other ants is stochastically taken into account. Indeed, the next component is selected according to a probability P_{lj} given by:

$$P_{lj} = \frac{\tau_{lj}^{\alpha_{ANT}} \eta_{lj}^{\beta_{ANT}}}{\sum_{k \in N_l} \tau_{lk}^{\alpha_{ANT}} \eta_{lk}^{\beta_{ANT}}} \quad (15)$$

Where N_l is the set of all possible paths for the solution component l (i.e., $|N_l| = K$), η_{lj} and τ_{lj} denote, respectively, the heuristic value given in equation (14), and the pheromone trail of the jth path for the flow originating from user l, and α_{ANT} and β_{ANT} determine, respectively, the relative importance of τ_{lj} and η_{lj}. Recall that η_{lj} represents the desirability of adding the solution component j (i.e., path j) to route the flow of user l, whereas τ_{lj} represents how proficient it has been so far to route the flow of user l through path j. As such,

α_{ANT} and β_{ANT} parameters have the following influence on the algorithm behavior. If $\beta_{ANT} = 0$, the selection probabilities are proportional to the heuristic value η_{lj}, which means that the components with high heuristic value are more likely to be selected. In this case, AC-OFER corresponds to a classical stochastic greedy algorithm. However, if $\alpha_{ANT} = 0$, only pheromone amplification is at work: the components with high pheromone trail are more likely to be selected, in which case a rapid convergence to a suboptimal solution may result as all ants are more likely to build the same solution.

On the other hand, in *exploitation*, the experience of the other ants is directly used. Indeed, among the possible components to add, the one with the highest value of $\tau_{lj}^{\alpha_{ANT}} \times \eta_{lj}^{\beta_{ANT}}$ is selected.

3) Selection of the Best Solution: The criterion to choose the best solution is the objective function given in equation (2), which takes into account the energy consumption, the on/off switching and re-routing costs.

4) Pheromone Trail Update: At the end of each iteration, the pheromones (trail values) for each flow l are updated as follows:

$$\tau_{lj} = (1 - \rho)\tau_{lj} + \Delta_{lj}^{best}$$

where $\rho \in [0, 1]$ is the decay coefficient of the pheromone, $\Delta_{lj}^{best} = Q/\eta^{best}$ if flow l is routed through the jth path in the best solution of the current iteration, 0 otherwise, and Q is a constant called the *pheromone update constant*. Recall that $\eta^{best} = 1/Objective\ function\ value\ of\ the\ best\ solution$, as reported in equation (14).

It is worth noting that when the score of the objective function is computed, we use the link rate adaptation provided in Algorithm 2.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the efficiency of our proposed approach. We first present the baselines we used for performance comparison as well as and the simulation parameters and environment. Then, we present detailed analysis of the simulation results.

A. Baselines

We compare the benefits of our AC-OFER approach with respect to four baselines: the Shortest Path (SP) routing, the Minimum link Residual Capacity (MRC) routing metric proposed in [13], the Load Balancing (LB) scheme, and Greedy-OFER. Note that the latter is similar to AC-OFER in the fact that it uses the same algorithm for routing new incoming flows. However, for network reconfiguration, it uses a greedy algorithm to find the solution instead of the Ant Colony algorithm. In other words, it seeks a feasible and acceptable solution by exploring the solution space and choosing the next step without iteration to improve the solution. This results in short computation times. Regarding the MRC baseline approach, the aim is to consolidate the traffic through the same paths in order to reduce the number of used nodes. Finally, LB is used to illustrate the worst case power consumption scenario.

Parameter	Value	Parameter	Value	Parameter	Value
α_A	0.12	A_{max}	5	$P_{linecard}$	2 W
β_A	1.1	N_{max}	8	$P_{r_{10}}$	4 W
q_0	0.1	P_{AG}	18 W	$P_{r_{100}}$	8 W
ρ	0.2	P_S	3 W	$P_{r_{1000}}$	10 W
		$P_{chassis}$	80 W	$P_{r_{10000}}$	15 W

TABLE IV
ENERGY SAVING COMPARISON WITH THE OPTIMAL SOLUTION

	Energy Saving	Computation time (ms)
Optimal	50.7%	4.8×10^5
AC-OFER	47.2%	381
Greedy-OFER	40.9%	178
MRC	33.4%	0
SP	9.7%	0
LB	0%	0

B. Simulation Environment

To evaluate our proposal, we developed our own discrete event simulator in *Java*. The simulator generates traffic and performs the routing according to the paths defined by the corresponding routing algorithms.

C. Simulation Parameters

Our analysis is based on random and tree-like topologies. However, due to space limitation, we present results only for tree-like campus network topologies. We considered different campus network sizes: small (≤ 100 APs), medium (100-200 APs), large (≥ 200 APs) and extra-large (≥ 2000 APs), with 1-8 gateway routers. As depicted in Fig. 1 and proposed in [10], the switches are divided into two groups: (i) Edge switches that connect the APs to the second layer composed of (ii) Aggregation/Core switches, which themselves are connected to the gateways.

In the wireless part, the interference range R_I of each AP is set to $1.5 \times R_t$, where R_t is the transmission range. The wireless links capacities are set to 54 *Mbps*. Note that in practice, and SDN controller can send frequent monitoring probes to estimate wireless link capacities using either active or passive measurement, such as the one proposed in [47]. For the users' arrival, we have used two scenarios. The first one is exploiting real traces provided by CRAWDAD [48]. Specifically, we have used a dataset that includes syslog records of user association/disassociation for several thousand users at Dartmouth College. The syslog record that indicates a user association is used as a new user arrival and its flow's lifetime is taken based on the corresponding syslog disassociation record. The second scenario models the users' arrival as a Poisson process with rate λ and an exponential lifetime of mean $1/\mu = 90$ minutes. Each user generates a flow with a uniform throughput demand between 1 and 10 Mbps in both uplink and downlink directions and a delay bound of 4 hops. Other simulation parameters are summarized in Table III and are based on works in [32], [36], [42] and [43]. It is worth noting that there is no optimal rule for

TABLE V
COMPUTATION TIME COMPARISON (IN MILLISECONDS)

	Small-sized network		Medium-sized network		Large-sized network		Extra-large-sized network	
	flow routing	Reconfig.	flow routing	Reconfig.	flow routing	Reconfig.	flow routing	Reconfig.
Optimal	25.3	4.8×10^5	-	-	-	-	-	-
AC-OFER	24.2	381	50.1	624	64.3	924	79.2	1134
Greedy-OFER	25.1	178	50	321	64.1	549	79.6	705
MRC	25.05	0	49.8	0	63.3	0	78.9	0
SP	23.7	0	49.3	0	63.1	0	79.0	0
LB	23.5	0	48.7	0	62.7	0	77.5	0

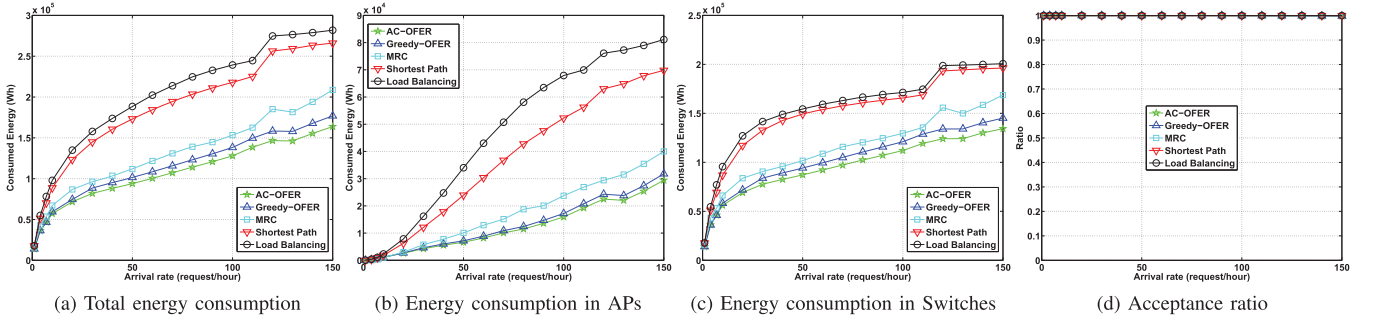| (a) Total energy consumption | (b) Energy consumption in APs | (c) Energy consumption in Switches | (d) Acceptance ratio |

Fig. 3. Comparison of energy consumption for variable arrival rates (100 APs, 27 switches with 2 gateway routers).

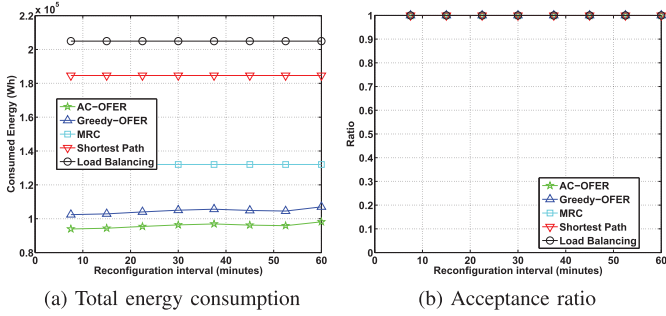| (a) Total energy consumption | (b) Acceptance ratio |

Fig. 4. Comparison of energy consumption for variable reconfiguration intervals (100 APs, 27 switches with 2 gateway routers, $\lambda = 50$ requests/hour).

setting the values of parameters β_{ANT}, α_{ANT}, ρ, q_0, the number of ants and the number of iterations, as pointed out in [49], [50]. Hence, we experimentally tuned these parameters by running preliminary tests using different values for each of them. More specifically, we vary β_{ANT}, α_{ANT}, ρ, q_0 between 0 and 1, by step of 0.05, and compare the objective function given in equation (2). We then pick the values that result in the smallest objective function presented in (2) (see Table III). In addition, since we focus on energy consumption, we set the parameters α_E to 0.9 and α_S, α_R to 0.05 each. Note that, for each network setup, Q is set to $\frac{1}{|L| \times O_{init}}$, where $|L|$ is the number of flows to route and O_{init} is the objective function score produced by any solution given by any other heuristic approach, as suggested by Dorigo et al. in [50].

The results are obtained over many simulation runs for each scenario, with a margin error less than 5%, then we calculate the average value of each performance metric. For sake of presentation, we do not plot confidence intervals.

In what follows, we first present the convergence of AC-OFER compared to the optimal solution and its computational

complexity of AC-OFER. Then, we present the impact of the arrival rate λ and the reconfiguration interval T for the case of small, medium and large-sized campus networks. Finally, we present results on the scalability of our approach in large-sized networks.

D. Convergence to the Optimal Solution and Computation Time

First, we show the convergence of our proposed approach towards the optimal solution given by the ILP presented in section IV. To do so, we develop a brute force algorithm that uses exhaustive search to find the optimal solution. As this problem is $\mathbb{NP}-$Hard, we run these tests only for small campus network topologies of 16 APs, and using real traces provided by CRAWDAD [48], for 48 hours. We measured the energy consumption of the different approaches compared to the optimal solution. The results are reported in Table IV. We can notice that AC-OFER achieves near-optimal solution, with only 3.5% decrease in energy saving on average compared to the optimal solution, but with much shorter runtime. On the other hand, compared to the greedy algorithm, AC-OFER achieves nearly 7% increase in the energy saving on average, but with higher computation time.

Table V further investigates the computation time required to find a new route for a new incoming flow as well as the reconfiguration time for all approaches in small, medium, large and extra-large-sized networks. We can notice that the time required to route a new incoming flow is almost the same for all approaches (around $24 - 25$ *ms* in the small-size case) since all approaches make use of the Dijkstra algorithm. However, the reconfiguration time of both Greedy- and AC-OFER are very short compared to the optimal solution. Indeed, more than 8 minutes are required for the optimal algorithm, while only
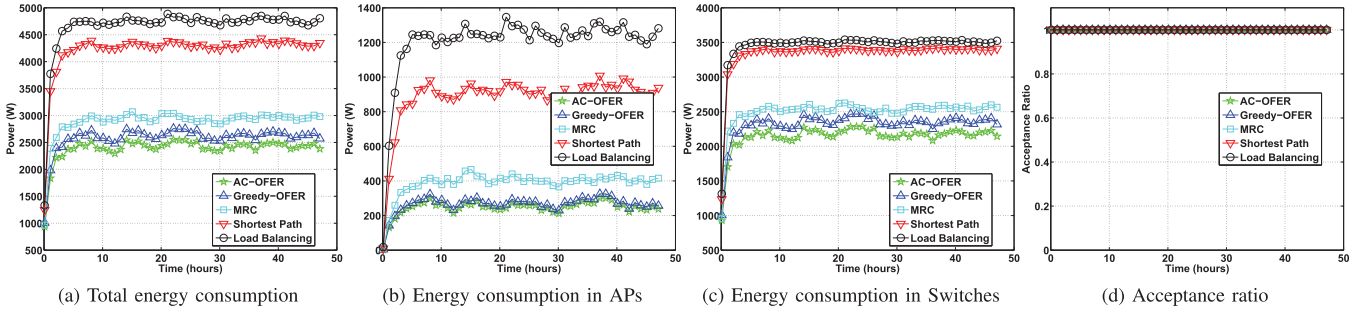
Fig. 5. Comparison of power consumption and acceptance ratio over time for $\lambda = 80$ requests/hour (100 APs, 27 switches with 2 gateway routers).

(a) Total energy consumption (b) Energy consumption in APs (c) Energy consumption in Switches (d) Acceptance ratio

381 *ms* and 178 *ms* are required for AC-OFER and Greedy-OFER, respectively. Note also that the computation time is always stable and AC-OFER stays tractable as the reconfiguration time is slightly over one second in the worst case (i.e. extra-large-sized networks). It is worth noting that, for these two latter schemes (i.e., Greedy- and AC-OFER), the computation time for each reconfiguration remains low compared to the reconfiguration period T, which is in the order of minutes (8 minutes and higher in our simulations). However, the optimal algorithm is clearly not suitable as the reconfiguration time is almost equal to the reconfiguration interval T.

E. Impact of Arrival Rate λ

Second, we study the impact of traffic load on our proposed approach. To do so, we vary the users arrival rate and measure the power consumption in the network for a simulation duration of 48 hours. Fig. 3(a), 3(b), 3(c) and 3(d) show, respectively, the total energy consumption for different arrival rates, the energy consumed by the APs and switches and the flow acceptance ratio in the case of medium-sized networks. From these figures, we can notice that:

- AC-OFER reduces the power consumption compared to the other schemes. Indeed, from Fig. 3(a), when $\lambda \in [10, 120]$, the power saving culminates at 10.5, 37, 100 and 120 *kWh* compared to Greedy-OFER, MRC, SP and LB, respectively. This corresponds to a power consumption decrease of approximately 6.5%, 17%, 42% and 45%, compared to Greedy-OFER, MRC, SP and LB, respectively. These gains are achieved in both the wireless part (i.e., APs) and the wired campus backbone, as shown in Figs. 3(b) and 3(c). Note that the energy consumption is reduced while the same acceptance ratio is realized for all schemes [see Fig. 3(d)].
- For low arrival rates (i.e., $\lambda < 10$), the power saving is negligible because of the light traffic load in the campus backbone. In fact, as there is few traffic in the network and spread around the whole network, flow consolidation is not always possible as the users are located in different areas and require turning on different APs and switches.
- For high arrival rates (i.e., $\lambda \geq 125$), the energy saving is stable. The reason behind this is that for high arrival rates, more capacity is needed mainly in the wireless part and flows can not be consolidated through the same paths due to APs capacity constraints.

F. Impact of the Reconfiguration Time T

Second, we study the impact of the reconfiguration interval T on the performance of AC-OFER. To do so, we fixed λ to 50 requests/hour and varied the reconfiguration interval T between 8 minutes and 1 hour. The total energy consumption and the acceptance ratio for the small-sized network case scenario are shown in Fig. 4.

We can observe from this figure that our approach outperforms the remaining solutions (i.e., Greedy algorithm, SP, MRC, and LB), especially in low values of T since frequent reconfiguration improves the flow re-routing and consolidation to achieve optimal energy consumption [see Fig. 4(a)]. Note that in these simulations, the same acceptance ratio is achieved in all approaches, as shown in Fig. 4(b).

G. Power Consumption Over Time

To further show the behavior of our approach over time, we plot in Fig. 5 the power consumption over time as well as the network utilization of all schemes for medium network loads (i.e., $\lambda = 80$ requests/hour). It is clear from this figure that the trend for AC-OFER is maintained over time. In fact, the total energy consumption as well as the energy consumption in the wireless and wired parts are maintained over time, as illustrated in Fig. 5(a), 5(b) and 5(c), respectively. Note that at the same time, the acceptance ratio is similar to all approaches, as shown in Fig. 5(d). More specifically, AC-OFER maintains the energy saving stable around 7%, 17%, 42% and 48% compared to Greedy-OFER, MRC, SP and LB, respectively.

To have a complete picture of the network performance, we plotted in Fig. 6 the normalized values of several performance metrics including acceptance ratio, total consumed energy, consumed energy by APs, consumed energy by switches, proportion of used APs, proportion of used switches, proportion of used links and average link utilization for used links. From this figure, we can observe that, AC-OFER accepts as many flows as SP, MRC and LB. However, it reduces at the same time the energy consumption in both APs and switches. This energy saving is achieved by reducing the number of used APs and switches. For instance, compared to LB, the gains are 48% for the total energy consumption, using 52% less APs and 31% less switches, respectively.

In addition, we can observe that our approach uses a reduced number of links compared to other schemes. In fact, AC-OFER reduces the proportion of used links by 4%, 15%, 43% and 52%
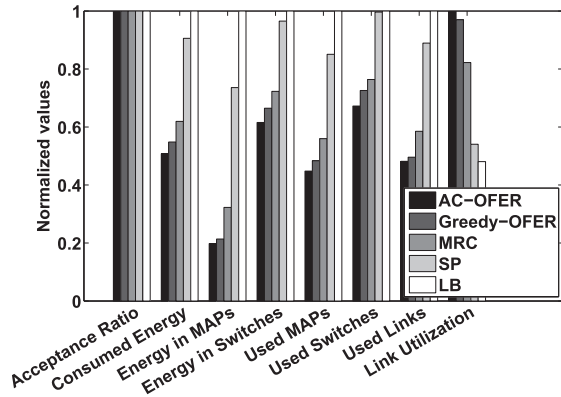
Fig. 6. Comparison of the average values of the different metrics (100 APs, 27 switches with 2 gateway routers, $\lambda = 80$ requests/hour).
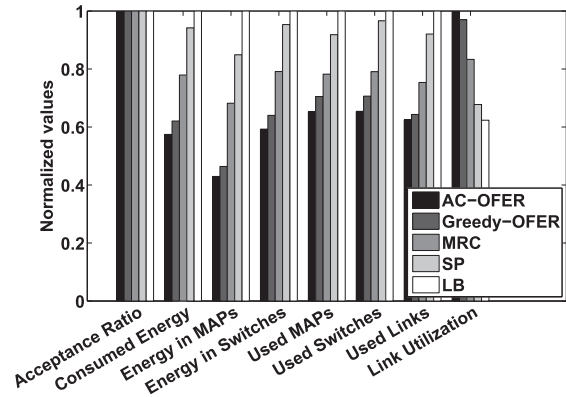

Fig. 7. Comparison of the average values of the different metrics (250 APs, 40 switches with 4 Gateways, $\lambda = 90$ requests/hour).

Fig. 8. Comparison of the average values of the different metrics (2000 APs, 280 switches with 8 Gateways, $\lambda = 350$ requests/hour).
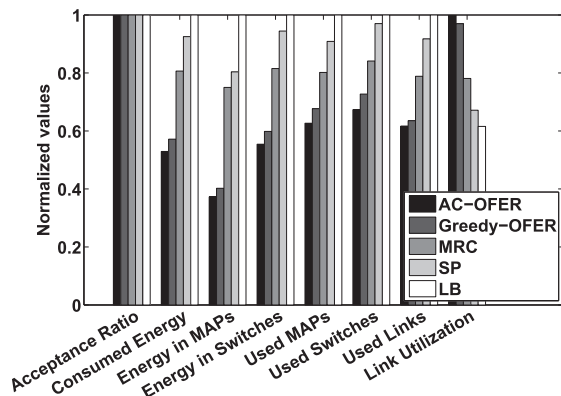
Fig. 8 shows the final results in the case of extra large campus networks. The same observation is made here too with AC-OFER achieving energy savings of 6.5%, 28%, 38% and 41% compared to Greedy-OFER, MRC, SP and LB, respectively. Similar remarks are also made as AC-OFER uses less APs, switches and links while presenting higher link utilization for the used links.

VII. CONCLUSION

In this paper, we investigated the energy efficiency problem in campus networks. We proposed an online flow-based approach that takes into account the dynamic arrival and departure of users by formulating the problem as an ILP and presenting an ant colony-based approach, called AC-OFER, to approximate the ILP optimal solution. Our objective is to minimize the energy consumption of the network, while routing dynamically the arriving and departing flows subject to QoS constraints (i.e., bandwidth and delay). Moreover, our approach uses link rate adaptation to further reduce energy consumption. Through extensive simulations, we showed that AC-OFER achieves significant reductions in terms of energy consumption, compared to the Greedy algorithm, the Shortest Path (SP) routing, the Minimum link Residual Capacity (MRC) routing metric and the Load Balancing (LB) scheme, while ensuring the required QoS. More specifically, we showed that AC-OFER can reduce the energy consumption by up to 7%, 35%, 44% and 49% compared to Greedy-OFER, MRC, SP and LB, respectively, for different network sizes and traffic loads. At the same time, AC-OFER guarantees a low time complexity for both route discovery and network reconfiguration. This approach represents therefore a promising solution for energy management in campus networks.

compared to Greedy-OFER, MRC, SP and LB, respectively. However, it results in high average link utilization of the used links due to flow consolidation. Indeed AC-OFER uses existing paths to route incoming flows, and performs the dynamic reconfiguration only at each time period T.

It is worth noting that MRC performs better than the LB and SP since, in this case, flows are consolidated according to the residual capacity. However, this scheme is clearly outperformed by AC-OFER thanks to the dynamic reconfiguration.

H. Scalability of AC-OFER

To study the scalability of our approach, we run additional simulations in the case of large-sized networks (i.e., 250 APs, 40 switches, 4 gateway routers) and extra large scale networks (i.e. 2000 APs, 280 switches and 8 gateways). Fig. 7 presents the final values (over 48 hours) of different metrics for the case of large networks. Similar performance results compared to small and medium networks are observed here. Indeed, the energy consumption is reduced by 7%, 35%, 44% and 49% compared to Greedy-OFER, MRC, SP and LB, respectively, while using a reduced number of APs and switches. In addition, the number of used links is reduced for AC-OFER compared to the other approaches. However, these links present higher link utilization.

REFERENCES

[1] M. P. Mills, "The cloud begins with coal—An overview of the electricity used by the global digital ecosystem," A Digital Power Group Report, Aug. 2013. [Online]. Available: http://goo.gl/ktp1l7

[2] Center for Energy-Efficient Telecommunications. (2013, Apr.). *The Power of Wireless Cloud* [Online]. Available: http://goo.gl/nkdajO

[3] "One net green campus network solution," Huawei Technologies White Paper, 2011.

[4] *Cisco Wireless Mesh Networking*, Cisco Enterprise Mobility 3.0 Design Guide, Ch. 10, 2013.

[5] Cisco, "Pioneering university deploys Canada's first 802.11n wireless network," Customer case study, 2008 [Online]. Available: http://bit.ly/1MVujP5

[6] Cisco White Paper, "Cisco visual networking index: Global mobile data traffic forecast update, 2011–2016," Cisco White Paper, Oct. 2012.

[7] Deploying Brocade PoE and PoE Solutions in the Campus Network, Broacade Solution Guide, 2011 [Online]. Available: http://bit.ly/1LB2uJh.

[8] K. Yap, T.-Y. Huang, B. Dodson, M. S. Lam, and N. McKeown, "Towards software-friendly networks," in *Proc. 1st ACM Asia-Pac. Workshop Syst.*, 2010, pp. 49–54.

[9] N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[10] ONF, "SDN in the campus environment," ONF Solution Brief, Oct. 2013.

[11] S. Ramanathan and E. L. Lloyd, "Scheduling algorithms for multihop radio networks," *IEEE/ACM Trans. Netw.*, vol. 1, no. 2, pp. 166–177, Apr. 1993.

[12] M. Kodialam and T. Nandagopal, "Characterizing achievable rates in multi-hop wireless mesh networks with orthogonal channels," *IEEE/ACM Trans. Netw.*, vol. 13, no. 4, pp. 868–880, Aug. 2005.

[13] P. Chowdhury, M. Tornatore, S. Sarkar, and B. Mukherjee, "Building a green wireless-optical broadband access network (woban)," *J. Lightw. Technol.*, vol. 28, no. 16, pp. 2219–2229, Aug. 2010.

[14] L. Chiaraviglio, M. Mellia, and F. Neri, "Reducing power consumption in backbone networks," in *IEEE Int. Conf. Commun.*, Jun. 2009, pp. 1–6.

[15] E. Gelenbe and C. Morfopoulou, "Power savings in packet networks via optimised routing," *Mobile Netw. Appl.*, vol. 17, no. 1, pp. 152–159, Feb. 2012.

[16] T. Mahmoodi, "Energy-aware routing in the cognitive packet network," *Perform. Eval.*, vol. 68, no. 4, pp. 338–346, Apr. 2011.

[17] S. Avallone and G. Ventre, "Energy efficient online routing of flows with additive constraints," *Comput. Netw.*, vol. 56, no. 10, pp. 2368–2382, 2012.

[18] B. Addis, A. Capone, G. Carello, L. G. Gianoli, and B. Sanso, "Energy management through optimized routing and device powering for greener communication networks," *IEEE/ACM Trans. Netw.*, vol. 22, no. 1, pp. 313–325, Feb. 2014.

[19] F. Francois, N. Wang, K. Moessner, and S. Georgoulas, "Optimizing link sleeping reconfigurations in ISP networks with off-peak time failure protection," *IEEE Trans. Netw. Serv. Manage.*, vol. 10, no. 2, pp. 176–188, Jun. 2013.

[20] B. Heller *et al.*, "Elastictree: Saving energy in data center networks," in *Proc. 7th USENIX Conf. Netw. Syst. Des. Implement. (NSDI'10)*, 2010, p. 17.

[21] W. Lin *et al.*, "GreenDCN: A general framework for achieving energy efficiency in data center networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 1, pp. 4–15, Jan. 2014.

[22] L. Wang, F. Zhang, C. Hou, J. A. Aroca, and Z. Liu, "Incorporating rate adaptation into green networking for future data centers," in *Proc. 12th IEEE Int. Symp. Netw. Comput. Appl. (NCA)*, 2013, pp. 106–109.

[23] A. Markiewicz, P. N. Tran, and A. Timm-Giel, "Energy consumption optimization for software defined networks considering dynamic traffic," in *Proc. IEEE 3rd Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2014, pp. 155–160.

[24] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen, "A survey of energy efficient network protocols for wireless networks," *Wireless Netw.*, vol. 7, pp. 343–358, Sep. 2001.

[25] A. P. Jardosh, K. Papagiannaki, E. M. Belding, K. C. Almeroth, G. Iannaccone, and B. Vinnakota, "Green WLANs: On-Demand WLAN Infrastructures," *Mobile Netw. Appl.*, vol. 14, no. 6, pp. 798–814, Dec. 2009.

[26] F. G. Debele, N. Li, M. Meo, M. Ricca, and Y. Zhang, "Experimenting resourceondemand strategies for green WLANs," *SIGMETRICS Perform. Eval. Rev.*, vol. 42, no. 3, pp. 61–66, Dec. 2014.

[27] M. A. Marsan, L. Chiaraviglio, D. Ciullo, and M. Meo, "A simple analytical model for the energy-efficient activation of access points in dense WLANs," in *Proc. 1st Int. Conf. Energy-Effic. Comput. Netw.*, 2010, pp. 159–168.

[28] T. D. Todd, A. A. Sayegh, M. N. Smadi, and D. Z. D. Zhao, "The need for access point power saving in solar powered WLAN mesh networks," *IEEE Netw.*, vol. 22, no. 3, pp. 4–10, May/Jun. 2008.

[29] F. Ganji *et al.*, "Greening campus WLANs: Energy-relevant usage and mobility patterns," *Comput. Netw.*, vol. 78, pp. 164–181, 2015, special Issue: Green Communications.

[30] T. Chen, Y. Yang, H. Zhang, H. Kim, and K. Horneman, "Network energy saving technologies for green wireless access networks," *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 30–38, Oct. 2011.

[31] A. Capone, F. Malandra, and B. Sansó, "Energy savings in wireless mesh networks in a time-variable context," *Mobile Netw. Appl.*, vol. 17, no. 2, pp. 1–14, 2011.

[32] B. Silvia, C. Antonio, and S. Brunilde, "Joint design and management of energy-aware Mesh Networks," *Ad Hoc Netw.*, vol. 10, no. 7, pp. 1482–1496, 2012.

[33] J. Luo, C. Rosenberg, and A. Girard, "Engineering wireless mesh networks: Joint scheduling, routing, power control, and rate adaptation," *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1387–1400, Oct. 2010.

[34] A. Amokrane, R. Langar, R. Boutaba, and G. Pujolle, "A green framework for energy efficient management in TDMA-based wireless mesh networks," in *Proc. 8th Int. Conf. Netw. Serv. Manage. (CNSM)*, Oct. 2012, pp. 322–328.

[35] A. Amokrane, R. Langar, R. Boutaba, and G. Pujolle, "Online flow-based energy efficient management in wireless mesh networks," in *Proc. IEEE GLOBECOM*, Dec. 2013, pp. 329–335.

[36] S. Chen and G. Muntean, "E-Mesh: An energy-efficient cross-layer solution for video delivery in wireless mesh networks," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2012, pp. 1–7.

[37] L. Budzisz *et al.*, "Dynamic resource provisioning for energy efficiency in wireless access networks: A survey and an outlook," *IEEE Commun. Surv. Tuts.*, vol. 16, no. 4, pp. 2259–2285, Fourth Quart. 2014.

[38] P. Dely, A. Kassler, and N. Bayer, "OpenFlow for wireless mesh networks," in *Proc. 20th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Aug. 2011, pp. 1–6.

[39] L. Suresh, J. Schulz-Zander, R. Merz, A. Feldmann, and T. Vazao, "Towards programmable enterprise WLANs with Odin," in *Proc. 1st Workshop Hot Topics Softw. Defined Netw. (HotSDN)*, 2012, pp. 115–120.

[40] Y. Al-Hazmi, H. de Meer, K. Hummel, H. Meyer, M. Meo, and D. Remondo, "Energy-efficient wireless mesh infrastructures," *IEEE Netw.*, vol. 25, no. 2, pp. 32–38, Mar./Apr. 2011.

[41] K. Gomez, R. Riggio, T. Rasheed, and F. Granelli, "Analysing the energy consumption behaviour of WiFi networks," in *Proc. IEEE Online Conf. Green Commun. (GreenCom)*, 2011, pp. 98–104.

[42] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, "A power benchmarking framework for network devices," in *Proc. Int. IFIP Netw. Conf.*, 2009, pp. 795–808.

[43] D. Kliazovich, P. Bouvry, Y. Audzevich, and S. Khan, "Greencloud: A packet-level simulator of energy-aware cloud computing data centers," in *Proc. IEEE GLOBECOM*, 2010, pp. 1–5.

[44] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, "Reducing network energy consumption via sleeping and rate-adaptation," in *Proc. 5th USENIX Symp. Netw. Syst. Des. Implement. (NSDI'08)*, 2008, pp. 323–336.

[45] N. Thanh *et al.*, "Modeling and experimenting combined smart sleep and power scaling algorithms in energy-aware data center networks," *Simul. Modell. Pract. Theory*, vol. 39, pp. 20–40, 2013.

[46] M. Dorigo and L. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.

[47] L.-J. Chen, T. Sun, G. Yang, M. Sanadidi, and M. Gerla, "Monitoring access link capacity using TFRC probe," *Comput. Commun.*, vol. 29, no. 10, pp. 1605–1613, 2006.

[48] D. Kotz, T. Henderson, I. Abyzov, and J. Yeo. (Sep. 2009, Sep.). *CRAWDAD Dataset Dartmouth/Campus (V. 2009-09-09)* [Online]. Available: http://crawdad.org/dartmouth/campus/20090909

[49] A. Siemiński, "Ant colony optimization parameter evaluation," in *Multimedia and Internet Systems: Theory and Practice*, Berlin, Germany: Springer, pp. 143–153, 2013.

[50] M. Dorigo and G. Di Caro, "Ant colony optimization: A new meta-heuristic," in *Proc. Congr. Evol. Comput. (CEC'99)*, 1999, vol. 2, pp.–1477.

Ahmed Amokrane received the M.Sc. degree in computer science from École Normale Supérieure de Cachan, Cachan, France, in 2011, and the Ph.D. degree in computer science from the Pierre and Marie Curie University, Paris, France, in 2014. He was a Visiting Ph.D. Student for 9 months at the University of Waterloo, Waterloo, ON, Canada, and a Research Intern for 6 months with IBM T.J Watson Research Center, Yorktown Heights, NY, USA. He is currently working as a Consultant and R&D Engineer with CoESSI, Paris, France. His research interests include energy efficiency, green networking, and security in wireless networks, cloud computing, and software-defined networks.

Rami Langar received the M.Sc. degree in network and computer science from the Pierre and Marie Curie University, Paris, France, in 2002, and the Ph.D. degree in network and computer science from Telecom ParisTech, Paris, France, in 2006. He is currently an Associate Professor with Laboratoire d'Informatique de Paris 6 (LIP6), Pierre and Marie Curie University. In 2007 and 2008, he was with the School of Computer Science, University of Waterloo, Waterloo, ON, Canada, as a Postdoctoral Research Fellow. His research interests include mobility and resource management in cloud radio access networks, wireless mesh, vehicular ad hoc and femtocell networks, green networking, green cloud, and quality-of-service support.

Raouf Boutaba received the M.Sc. and Ph.D. degrees in computer science from the University Pierre and Marie Curie, Paris, in 1990 and 1994, respectively. He is currently a Professor of Computer Science at the University of Waterloo, Canada. His research interests include resource and service management in networks and distributed systems. He is the Founding Editor-in-Chief of the IEEE Transactions on Network and Service Management (2007–2010), and on the editorial boards of other journals. He has received several best paper awards and other recognitions such as the Premiers Research Excellence Award, the IEEE ComSoc Hal Sobol, Fred W. Ellersick, Joe LociCero, Dan Stokesbury, Salah Aidarous Awards, and the IEEE Canada McNaughton Gold Medal. He is a fellow of the IEEE, the Engineering Institute of Canada, and the Canadian Academy of Engineering.

Guy Pujolle received the Ph.D. and "These d'Etat" degrees in computer science from the University of Paris IX and Paris XI, in 1975 and 1978, respectively. He is currently a Professor with Pierre et Marie Curie University-Paris, 6, a Distinguished Invited Professor with POSTECH, Korea, a Member of the Institut Universitaire de France, and a Member of The Royal Physiographical Academy of Lund, Sweden. He spent the period 1994–2000 as Professor and Head of the Department of Computer Science, Versailles University. He was also a Professor and Head of the MASI Laboratory, Pierre et Marie Curie University (1981–1993), Professor with ENST (1979–1981), and a Member of the Scientific Staff of INRIA (1974–1979). He is the French representative at the Technical Committee on Networking at IFIP. He is an Editor for *ACM International Journal of Network Management, Telecommunication Systems*, and Editor in Chief of *Annals of Telecommunications*. He is a pioneer in high-speed networking having led the development of the first Gbit/s network to be tested in 1980. He has participated in several important patents like DPI or virtual networks. Guy Pujolle is Co-founder of QoSMOS (www.qosmos.fr), Ucopia Communications (www.ucopia.com), EtherTrust (www.ethertrust.com), Virtuor (www.VirtuOR.fr), and Green Communications (www.green-communications.fr).