Mobile Netw Appl
DOI 10.1007/s11036-016-0783-8

Policy-Based Management for Green Mobile Networks Through Software-Defined Networking

Alberto Huertas Celdrán[1] · **Manuel Gil Pérez**[1] · **Félix J. García Clemente**[2] · **Gregorio Martínez Pérez**[1]

Abstract Traditional networks are characterized by wasting considerable amount of energy that could be reduced drastically. The challenge of energy saving should be managed efficiently, where the mobility of users and services are nominated to play a significant role as well as the use of the Software Defined Networking (SDN) paradigm. Besides the network management supported by the SDN paradigm, we highlight the management of the network infrastructure at run-time, considering aspects like the energy efficiency. In this paper, we present an energy-aware and policy-based system oriented to the SDN paradigm, which allows managing the network infrastructure dynamically at run-time and on demand through policies. With these policies, any network using our solution will be able to reduce energy consumption by switching on/off its resources when they are inefficient, and creating virtualized network resources like proxies to reduce the network traffic. The experiments conducted demonstrate how the energy consumption is reduced when enforcing the proposed policies, considering aspects such as the number of base stations, their cell sizes, and the number of active devices in a given time, among other.

Keywords Energy-aware policies · Software defined networking · Users' mobility · Virtualization

1 Introduction

Energy saving is a critical task in the design and effective management of any computer network. The energy-aware challenge has been a cornerstone in traditional networks, in which the whole infrastructure needs to consume energy to provide users with services. All these services are invariant over time, because the network does not take into account aspects such as the current load traffic, the number of active devices, the date and time, or the patterns of the users' behavior. This fact implies the inefficient use of the network infrastructure and therefore the waste of energy.

The *Software Defined Networking* (SDN) paradigm has recently emerged with the main goal of easing network management [1, 2]. This paradigm allows network administrators to configure and manage their network status at run-time and on demand, where the energy efficiency should be considered in addition to other areas of application such as the ones studied in [3, 4]. Despite the number of resources and facilities provided by the SDN paradigm, the mobility provided by current devices and services has hindered the management of the network infrastructure efficiently. Nowadays, mobile networks are not used at their full capacity, except in given peak times. This entails a high waste of energy in situations with low traffic, while the infrastructure remains optimized for maximum traffic load.

✉ Alberto Huertas Celdrán
alberto.huertas@um.es

Manuel Gil Pérez
mgilperez@um.es

Félix J. García Clemente
fgarcia@um.es

Gregorio Martínez Pérez
gregorio@um.es

[1] Departamento Ingeniería de la Información y las Comunicaciones, University of Murcia, 30071, Murcia, Spain

[2] Departamento Ingeniería y Tecnología de Computadores, University of Murcia, 30071, Murcia, Spain

Important improvements in terms of energy saving can be achieved by managing the network resources at run-time, considering the mobility of users and services. In that sense, this paper at hand presents a mobility-aware policy-based system in charge of reducing the energy consumption in networks oriented to the SDN paradigm. The policies defined in our solution allow the SDN paradigm to switch on/off network resources when they are consuming energy in an inefficient way, as well as create virtualized network resources like proxies to reduce the network traffic generated by users consuming services close to the network infrastructure.

The remainder of the paper is structured as follows. Section 2 discusses the related work of other energy-aware solutions. A motivating example is presented in Section 3, which will be used through the paper at hand to introduce all concepts related to our proposal. The energy-aware policies and how they manage the energy consumption in our use case are presented in Section 4. Section 5 shows the ontology to shape the information as well as the technologies for the proposed architecture. Section 6 reports some experimental results to illustrate the performance of our proposal. Finally, conclusions and future work are drawn in Section 7.

2 Related work

A recent survey can be found in the current literature, in which a deep comparison of a number of green mobile network approaches was performed [5]. In this paper, the authors recognized that more than the 50 % of the energy consumed by mobile networks was produced in base stations (BS). In order to reduce this consumption, the analyzed solutions were classified into five different categories: improvement on the hardware components, sleep mode techniques, optimization in the radio transmission process, network planning and deployment, and adoption of renewable energy resources.

From the *energy efficiency* perspective, the solution presented in [6] quantifies power consumption of mobile communication systems. It establishes that there is a high potential to reduce the energy consumption when improving the energy efficiency of the BSs at low traffic load. Another proposal to enhance efficiency of power amplifier for wireless BSs was proposed in [7]. On the other hand, the second category was focused on *sleep mode* techniques, which turn on/off network resources during non-peak traffic hour selectively. The variation of the traffic patterns over time, in order to decide when BSs should sleep or not, was considered in [8]. Another solution was proposed in [9], where a control mechanism enables small cells to switch off all components while not serving active connections. In order to speed up the decision process of switching on/off BSs, a transfer actor-critic algorithm (TACT) was proposed in [10], making use of historical data from neighbor regions.

The *cell zooming* solution proposed in [11] provided a level of flexibility higher than the previous sleep mode proposals. This solution was capable of adjusting cell sizes according to different aspects such as the traffic load, the users' requirements, or the channel conditions. It allowed balancing the traffic load by zooming in/out the cell, in order to reduce/increase the coverage area to avoid congestion. In [12], it was studied the effect of the cell size on the energy consumed by BSs that make use of current technologies. Its authors recognized that the optimal cell size from an energy perspective depends on several factors such as the technology of the BS, the data rates, and the traffic demands. They also proposed a schema to adjust the cell sizes dynamically for saving energy. On the other hand, the concept of area power consumption was introduced in [13], in order to evaluate the impact of deployment strategies on power consumption of mobile radio networks. They considered different numbers of micro BSs, in addition to the conventional macro sizes. Their outcomes indicated that it had a moderate effect on power consumption by using micro BSs in scenarios with full traffic load.

Regarding the *radio transmission* process, a number of approaches have been proposed to efficiently make use of resources in time, frequency, and spatial domains. The multiple network access interfaces of the mobile devices were used in [14], with which they can cooperate in sending data packets to the BS. Another solution oriented to cognitive radio transmission was proposed in [15], where a multi-hop cognitive cellular network architecture facilitates the ever exploding data transmissions in cellular networks. In this same research area, a comparative study of the energy consumption of several wireless network access points can be found in [16].

Deploying low power-consuming small cells (micro, pico, and femto cells) in reduced areas with dense traffic was proposed in [17]. It analyzed the energy efficiency in small cell networks, using a random spatial network model in which BSs and users were modeled as two independent spatial Poisson Point Processes (PPP).

Finally, the last category identified in [5] included several approaches adopting *renewable energy resources*. Among them, in [18] it was proposed an energy cooperation model between cellular BSs with hybrid energy sources, limited storage, and a connecting power line.

So far, we have seen solutions oriented to reduce the energy consumption in BSs. However, data centers are also consuming huge amount of energy for computing. Many designs in data centers rely on virtualization to eliminate the hardware constraints and make the computation more flexible and efficient. In this context, a mechanism able to transfer virtualized resources from one physical machine to another was proposed in [19], without interrupting its services. In other solutions, as the one proposed in [20], an event-driven network control framework that used high-level policies was presented to configure and manage the network state, which are translated into a set of forwarding rules to be managed by the controller.

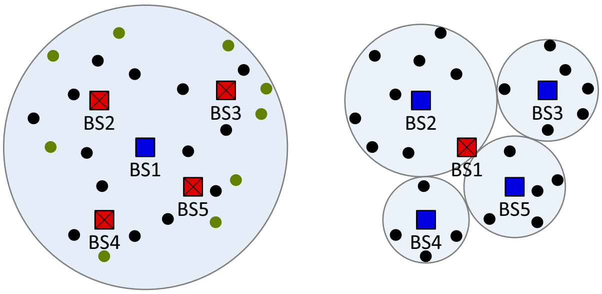
As commented on earlier, the previous solutions use different metrics and techniques in order to save energy or reduce the energy consumption. In [21], it was presented a survey where a high number of green proposals show researches, experiments, deployments, and evaluations of green metrics for mobile networks. These metrics can be categorized into two types: *equipment-level* and *facility-level* metrics. Equipment-level metrics are focused on the lower-level energy efficiency. They consider individual pieces of mobile networks such as, for example, Energy Consumption Rating (ECR), Consumer Consumption Rations (CCR), and Telecommunications Energy Efficiency Ratio (TEER). Instead, facility-level metrics are related to high-level energy efficiency. They are focused in a macro point of view, such as data center metrics, including Power Usage Effectiveness (PUE), Data Center Infrastructure Efficiency (DCiE), and Data Center Productivity (DCP).

Despite the previous solutions improved energy efficiency with novel techniques and metrics, the decision of applying them was conducted by administrators. In that sense, our solution is able to decide at run-time and on demand when the appropriate techniques should be applied by using our energy-aware policies.

3 Use case in mobile scenarios

In this section, we present a use case to illustrate how our solution manages the energy efficiency at run-time in networks providing services to devices with a high mobility oriented to the SDN paradigm. The proposed use case is composed of five BSs distributed along a given area. Among them, one (BS1) is providing devices with different services, whereas the rest are asleep (BS2, BS3, BS4, and BS5) for saving energy. Figure 1a shows a given time when new devices, marked as green (lighter) points, appear in the BS1's cover area, and start consuming services and move away from BS1. Its cell size is then automatically increased to continue providing services. This situation generates an energy-aware concern, because BS1 needs to consume more energy in increasing its cell size for providing the services.

In order to fix this problem, our solution switches on the asleep BSs located close to the mobile devices and switches off BS1, as shown in Fig. 1b. Despite of switching on new BSs, they consume less energy than BS1. This is because users are close to them and their coverage cells are smaller than the one needed by BS1. (It is worthy to note that this assertion in energy consumption is later demonstrated in Section 6.) The sizes of the coverage cells are computed by the network infrastructure dynamically, using for example the load balancing technique of [11]. This technique automatically increases or decreases the size of BSs' cells by considering the energy efficiency. It is worth noting that if users go back closer to BS1, or even the number of active devices consuming services decreases, our solution will switch on BS1 and switch off the rest of BSs.



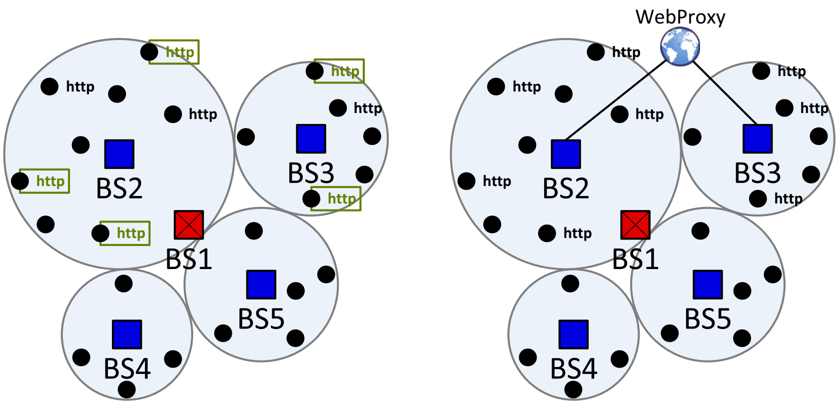
(a) Initial scenario (b) Switched ON/OFF scenario

Fig. 1 Base stations (BS) are switched on/off to consume energy in an efficient way. *Blue* (not strikethrough) squares represent BSs switched on, *red* (strikethrough) squares switched off, *black* (darker) circles are devices consuming services, and *green* (lighter) circles newcomers

Following the use case, Fig. 2a shows the moment when several active devices located at the cover areas of BS2 and BS3 start visiting certain websites, thereby consuming http services. This situation generates a new energy-aware concern, because the network infrastructure has to transmit and receive packages. In order to reduce this energy consumption, our solution is capable of deploying virtualized resources in the network. A *virtualized web proxy* is an example of virtualized infrastructure that allows temporary storage (caching) of information. For example, virtualized web proxies can cache web pages to reduce network traffic in future http requests, as it will not be necessary to retrieve again that information from the web service in future requests. Reducing network traffic will entail a reduction in energy consumption, both for the devices and for the network infrastructure [22]. Within our use case, Fig. 2b shows how a virtualized web proxy is created close to BS2 and BS3, in order to provide active devices of BS2 and BS3 with the http service. The reverse process (dismantling the virtualized proxy) is performed by our solution when users stop consuming that service.

4 Energy-aware and management-oriented policies

Our solution allows the SDN paradigm to manage at run-time the energy consumption of the network infrastructure



(a) Initial scenario (b) Managed scenario

Fig. 2 Virtualized web proxies are created/dismantled to reduce energy consumption. *Blue* (not strikethrough) squares represent BSs switched on, *red* (strikethrough) squares switched off, *black* (not boxed) text makes reference to devices consuming http services, and *green* (boxed) text to devices starting to consume http services

by using policies. Among the different existing policies, we make use of energy-aware and management-oriented policies. The network administrator is in charge of defining the set of policies that will decide the list of potential actions to be taken by the SDN components, in accordance with the energy consumption, the users' mobility, and the network statistics.

Our policies are comprised by the following elements: the *type* of policy (e.g., switching, virtualization, etc.); the network *resource*, whose information is currently being managed (e.g., base station, switch, service, etc.); the *metric* with which the network state can be evaluated (e.g., performance indicators like PI_{urban} [23] or ABf, which indicate the average power consumption in peak hour and the Average of Bytes per flow, respectively); the *location* or region where the policy will be enforced (e.g., geographic position, area, etc.); the *date* or the period of time at which the policy will be applied (e.g., hour, timestamp, etc.); and the *result* or set of actions to be carried out over the network once the policy is applied (e.g., switch, create a proxy, etc.).

The previous set of elements defining a given policy can be represented as follows.

$$Type \wedge Resource \wedge Metric \wedge Location \wedge Date \rightarrow Result$$

4.1 Policies to ensure the energy efficiency

We introduce below the two kinds of policies required to manage the energy consumption of the previous use case. It is worthy to note that other policies could also be defined, since the proposed solution is extensible.

4.1.1 Switching policies

These policies allow the SDN paradigm to switch on/off the network resources located at specific locations. By using this kind of policies, the network administrator can indicate that when a given network *resource* is making an inefficient use of the energy (e.g., the PI_{urban} *metric* is below a given threshold, which was defined by the network administrator beforehand), the *result* is to switch on or off the network elements whose *location* is close to the inefficient ones.

In addition, this kind of policies can also be applied in a proactive way, in case of knowing the patterns of energy consumption or the users' mobility. In switching policies, the *metric*, *location*, and *date* parameters are optional. It is important to note that once the network resources are switched on/off, the network traffic should be balanced in the network infrastructure.

4.1.2 Virtualization policies

These policies allow creating or dismantling the virtualized network resources to optimize the energy consumption of the network infrastructure as well as the devices when users consume certain services.

The virtualization policies can save energy in many different ways. For example, virtualized infrastructure can be created to balance the load traffic and optimize the use of the resources, or even in charge of reducing the network traffic and thus the time to consume services. By using these policies, the network administrator can indicate that when a given network (*resource*) is making an inefficient use of the energy (e.g., PI_{urban} *metric* is below another threshold), the *result* is to create a given number of virtualized resources whose *location* is close to the inefficient ones. As before, the *metric*, *location*, and *date* parameters are optional in this kind of policies.

4.2 How to reduce energy consumption in mobile scenarios

How our solution can be used to reduce the consumed energy is explained below, in which we use the use case presented in Section 3.

In order to switch on/off the network resources, as depicted in Fig. 1b, our solution defines two generic *switching policies*. They are shown below as an example. These policies indicate that when the *PIUrban* value of any BS is within the *AlarmA* range values (the range of this alarm is set by the network administrator), the network should switch on the BSs located at the same area that the inefficient one and switch off the latter. It is important to note that, despite these examples of policies use the metric and location parameters, they are optional in our policies.

The policy shown below is in charge of switching on the BSs located close to the inefficient one, as shown in Fig. 1 as an example of a proof of concept.

```
Type(#Switching) ∧ BaseStation(?bs) ∧
Location(?bs,?area) ∧ locatedBS(?area,?neighborBSs) ∧
integer[PIUrban in #AlarmA] hasPIUrban(?bs)
→ hasStatus(?neighborBSs,#ON)
```

Once the previous policy is enforced, the next action is to implement the following policy in order to switch off the inefficient BS.

```
Type(#Switching) ∧ BaseStation(?bs) ∧
integer[PIUrban in #AlarmA] hasPIUrban(?bs)
→ hasStatus(?bs,#OFF)
```

In our use case, these two policies switch on BS2, BS3, BS4, and BS5 and switch off BS1. It is important to note that the cell sizes of the new active BSs are calculated by the network infrastructure, by making use of balancing techniques like the one proposed in [11].

Following the use case proposed in Section 3, we have defined two *virtualization policies* to reduce the energy consumption in the network infrastructure and in the users' devices. The first virtualization policy is in charge of creating and associating virtualized web proxies to reduce the traffic of the http service in BSs located close to the inefficient ones. This inefficiency is measured in our policy by using the values of the PI_{urban} and ABf metrics. Specifically, when the values of these two metrics are within the *AlarmB* and *AlarmC* range values, a web proxy will be created (if not any) and it will be associated to the BS.

```
Type(#Virtualization) ∧ BaseStation(?bs) ∧
WebProxy(?webP) ∧
integer[PIUrban in #AlarmB] hasPIUrban(?bs) ∧
integer[ABf in #AlarmC] hasABf(#HttpService)
→ associate(?bs,?webP)
```

Following the use case, the previous policy creates a virtualized web proxy associated to BS2 and BS3, in order to reduce the http traffic generated by the users located in the cover areas of these BSs.

Finally, the web proxy is disassociated once users stop consuming the http service, as well as being dismantled if it is not used anymore. This is shown below by the next policy.

```
Type(#Virtualization) ∧ BaseStation(?bs) ∧
WebProxy(?webP) ∧
integer[PIUrban in #AlarmD] hasPIUrban(?bs) ∧
integer[ABf in #AlarmE] hasABf(#HttpService)
→ disassociate(?bs,?webP)
```

5 Policy deployment infrastructure

This section shows how our solution models the network information through ontologies. Moreover, we also propose an architecture with which to manage the ontology proposed in this section as well as the policies defined in Section 4, with the aim of saving energy in network resources and users' devices.

5.1 The mobile network ontology

We describe here the main ontology that is managed by our solution, which models all the concepts introduced along the paper. With this ontology, we provide a set of primitives with which to describe a collection of the elements that belong to the mobile network topic and the relationships among them. This ontology is shown in Fig. 3, which is focused on shaping the information of the base stations and the devices. All that information is shaped in OWL 2 (Web Ontology Language) [24], using Protégé [25] for its generation.

The mobile network ontology is categorized into two different, but related topics: base station and device. The top-level class in the former topic is *BaseStation*, which is one of the components where we reduce the energy consumption in mobile networks. A base station is a kind of the network *InfrastructureElement*, which is composed of several *Components*, and it can in turn use different *Technologies* for its design. As a proof of concept, we have modeled four of the main components that compose any kind of base station.

It is worthy to note that all elements commented so far, and the ones introduced below, have been modeled by following the Common Information Model (CIM) language defined by the DMTF as a standard [26]. This language provides a common definition of management information for systems, networks, applications, and services. Due to that, all the elements herein defined inherit from the *ManagedElement* class, defined in CIM as the main element for a given model.

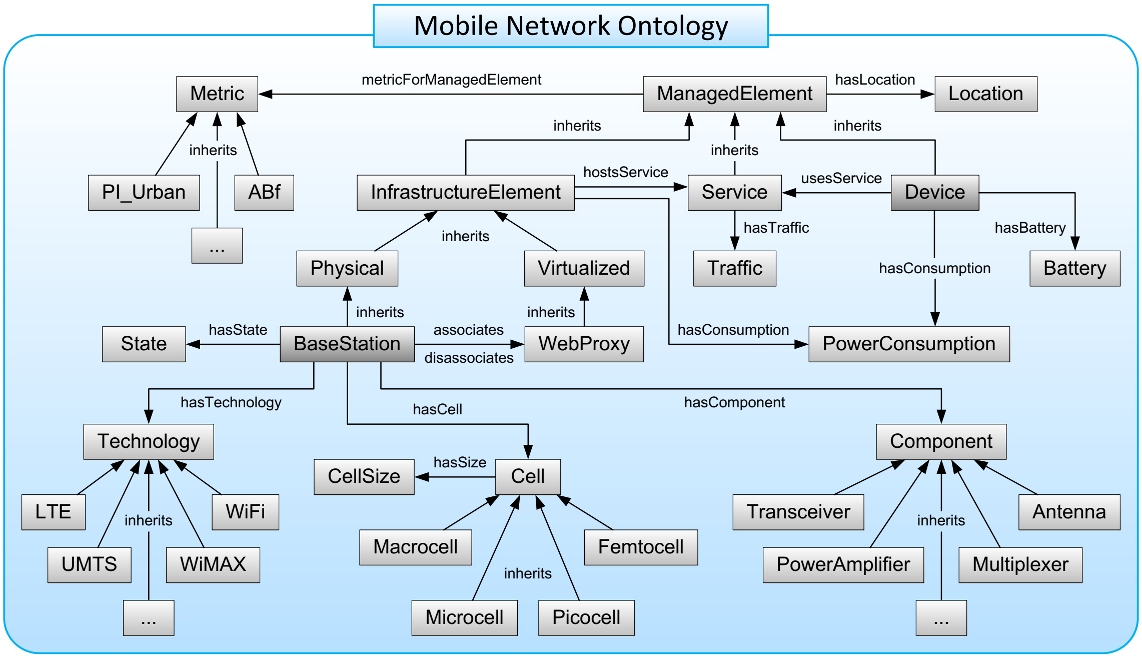


Fig. 3 Elements of the proposed ontology for shaping a mobile network

The Component class has four predefined subclasses: *Antenna* makes reference to the structure for sending and receiving electromagnetic waves; *PowerAmplifier* in charge of amplifying the signal for any transmission; *Multiplexer* modeling ways of separating sending and receiving signals; and *Transceiver*, which does transmission and reception of signals. Nowadays, we can find several technologies of base stations, such as *LTE*, *UMTS*, *WiMAX*, *WiFi*, etc. Considering the size of the *Cells*, named *CellSize* in Fig. 3, base stations can be *MacroCell*, *MicroCell*, *PicoCell*, and *FemtoCell*. Moreover, base stations are located at specific positions that are modeled by the *Location* class. Finally, the energy consumed by base stations is considered by the *PowerConsumption* class.

The another topic in our mobile network ontology is the device, where the *Device* class is the main element. This is the other point in which our solution is focused on for saving energy through policies. All devices are connected to base stations with the aim of consuming the *Services* provided by the network. Furthermore, as the base stations, devices are located at specific position and consume energy. These two aspects are modeled by *Location* and *PowerConsumption*, respectively, which have been commented earlier. From the energy saving perspective, we have modeled *Battery* as an example.

All entities defined earlier for the mobile network ontology are related each other by properties, where a portion of them is used to establish new relationships through policies. For example, the *BaseStation* class use the *hasTechnology*, *hasCell*, and *hasComponent* properties for defining how a given base station is implemented internally. With respect to the another category of our mobile network ontology, the *Device* class is linked to *Service* and *Battery* via the *usesService* and *hasBattery* properties, respectively, in order to know the services that are being used by each device as well as its corresponding battery for energy issues. Note that all these elements are connected to *PowerConsumption*, through the *hasConsumption* property, so as to know the energy consumed by each base station and by each device.

5.2 Deployment of our energy-aware and policy-based system

This section shows the architecture of our energy-aware solution in charge of managing mobile networks oriented to the SDN paradigm at run-time. Figure 4 shows this architecture, where the *SDN plane* contains the layers of the SDN paradigm and the *SDN management plane* depicts the components shaping our solution.

From bottom to top, the first component of our proposal is the *Collector*. This is in charge of joining the infrastructure location, gathered from an independent *Location Middleware*, with the network configuration and statistics received from the *Controller*. It is important to know that the network configuration contains energy-aware information, such as power consumption or cell sizes. All this information is obtained by our OpenDaylight controller [27], using southbound protocols like OpenFlow or SNMP.

When the *Interpreter* component receives the joined information from the *Collector*, the former stores all this

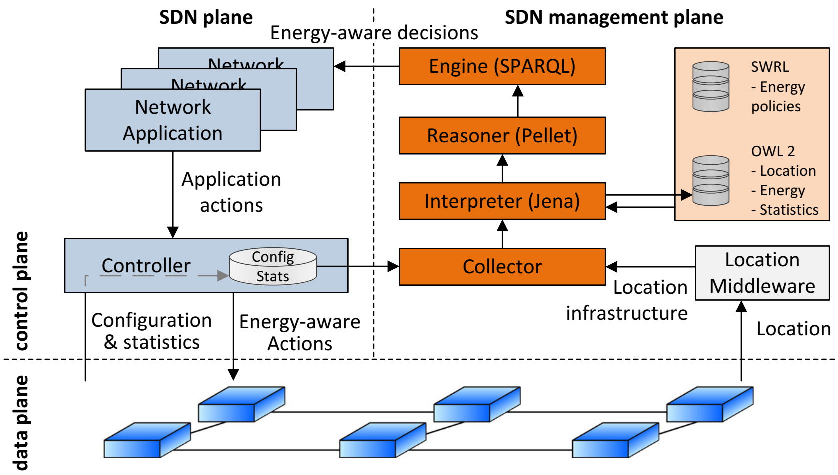


Fig. 4 Architecture of the proposed mobility-aware and policy-based system for energy saving

information by using the ontology detailed earlier. Once it is stored, the *Interpreter* uses the Jena API to generate ontological models with our ontology and the energy-aware policies defined in SWRL [28]. On the other hand, the *Reasoner* component is making use of Pellet [29], which receives the ontological models being generated by the *Interpreter* component and returns the inferred models with new knowledge.

Finally, the *Engine* is in charge of translating the decision(s) performed into SPARQL queries [30], which are applied to the inferred model. Therefore, the energy-aware results of the SPARQL queries are sent to the SDN applications, which will enforce the corresponding energy-aware actions over the network.

6 Experiment results

In this section, we conducted some experiments with the aim of demonstrating the usefulness of our solution. We modeled the energy consumption that is needed for offering a service to a given area A. This requires to consider the consumption of the base stations as well as the consumption derived from the active devices. To this end, we have used Eq. 1, proposed in [12], in order to obtain the energy consumption of the network.

$$P_{total} = c_1 \cdot N_{bs} + c_2 \cdot N_d \cdot d^e \tag{1}$$

where N_{bs} is the number of base stations; N_d represents the number of active devices; d is the cell size of the base station; c_1 and c_2 are related to the power consumption of the base stations, which depend on the technology used by those base stations; and e is the associated path-loss exponent. Moreover, it is important to take into account that if we want to cover an area A, the cell size of the base station (d) depends on the number of base stations: $N_{bs} = \frac{A}{d^2}$.

In order to calculate the power consumption of the network, we consider that the distance between the active devices and the base stations is d. In other words, we consider the worst case for the devices' location into the cells.

Considering the previous analysis, we are going to perform some experiments in order to see how the parameters of Eq. 1 affect to the network consumption. Specifically, Fig. 5 shows the variation of the power consumption for different fixed consumptions (c_1) when we increase the number of BSs (N_{bs}).

We can see with this experiment that, when the fixed consumption (c_1) decreases, it is better to have many base stations with small cells than a few of them with big cells. In Fig. 5, we can observe this situation when the fixed consumption is 2000W, whose optimum number of base

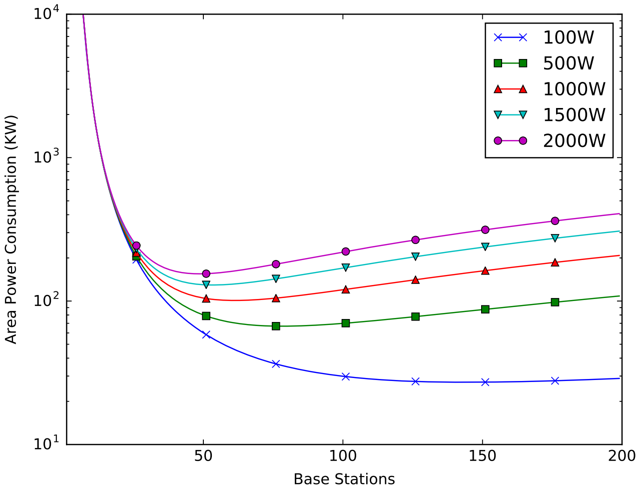


Fig. 5 Area power consumption modifying the number of base stations that cover a given area of 100 sq Km. Different curves show the fixed BS power consumption (c_1) when $c_2 = 0.5mW$ and $N_d = 10,000$

stations is close to 50, while this optimum number is close to 150 when the fixed consumption decreases to 100W.

Our next experiment is depicted in Fig. 6, where it is shown the variation of power consumption regarding the number of active devices and base stations.

Figure 6 demonstrates that when the number of active devices increases, several small cells are more efficient from an energy saving perspective than having a few of them with big cells. For example, if our network has 500 active devices, the optimum number of base stations is close to 20. However, in case the number of active devices increases to 50,000, the optimum number of base stations is between 100 and 150.

Based on the consumption energy model, and as it has been shown in the previous experiments, the number of active devices and the consumption of the base stations determine an optimum number of these base stations offering their services in each time. Our solution with a number of well-defined policies can switch on/off the base stations automatically in an optimum number of them. In Fig. 7, we show the specific points or moments (black circles) when our solution changes the network configuration to save energy.

In this figure, we can see how our solution changes the network configuration by using the policies defined in Section 4 to reduce energy consumption. PI_{urban} establishes the moment when the policies will change the network configuration. In this sense, Fig. 7 shows that when the PI_{urban} value is bigger than 2.5KW (64 active devices) and smaller than 5KW (758 active devices) our policies change the network configuration to 2 base stations. If the power consumption is increased between 5KW (758 active devices) and 11KW (3,693 active devices), it is better to have four base stations. Finally, if the energy consumption is higher than 11KW (we need over 3,693 active devices) the most suitable configuration consists on having 8 base stations.

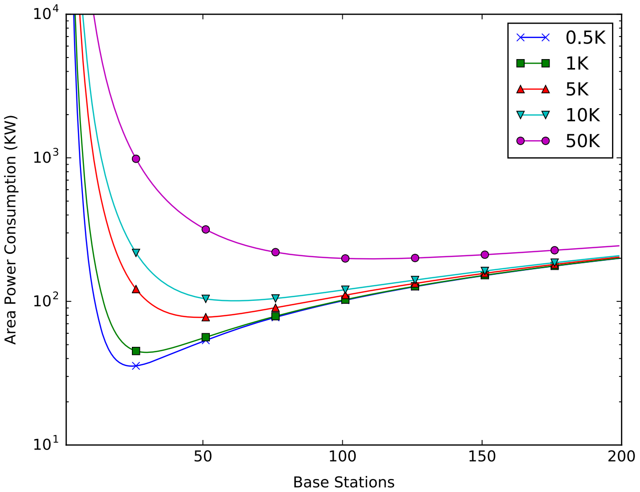


Fig. 6 Area power consumption modifying the number of base stations that cover a given area of 100 sq Km. Different curves show the number of active devices (N_d) when $c_1 = 1000W$ and $c_2 = 0.5mW$

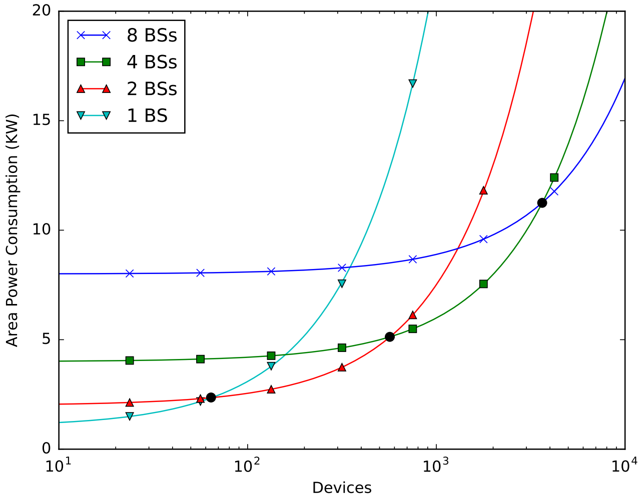


Fig. 7 Area power consumption modifying the number of active devices in a given area of 4 sq Km. Different curves represent the number of base stations when $c_1 = 1000W$ and $c_2 = 0.5mW$

In this section, we have demonstrated the usefulness of our solution, which is capable of changing dynamically and at run time the network configuration in order to consume energy in an optimal way. To this end, we have considered the fixed power consumption, the number of active devices, and the number of base stations and their cell sizes.

6.1 Decisions time

The Reasoner and Engine components are important parts of our solution. They are in charge of maintaining updated the information of the network infrastructure and deciding when our solution has to change the network configuration. In order to check the time required to make decisions at real time for reducing the energy consumption of the network, we conducted several experiments altering the complexity of our ontology.

We have increased the number of statements hold in the knowledge base, which depends on the number of individuals present in the ontology and the number of policies. Increasing the number of individuals and semantic rules, we will provoke an increment on the number of statements, and thus on the complexity of the executions. The conducted tests were carried out in a dedicated PC with an Intel Core i5-3317-U 3.40 GHz, 6 GB of RAM, and a Windows 10 as operative system. The results shown in this section have been obtained by executing the experiments 100 times and computing their arithmetic mean.

Table 1 Individual distribution of population

Element	Amount	Percentage
Devices	9,750	65 %
Location	4,500	30 %
Services	300	2 %
Others	300	2 %
Base Stations	150	1 %
Total	15,000	100 %

The number of individuals contained in our ontology is referred as *population*. This was randomly generated for the experiments, but in a controlled way in order to achieve the desired distribution for simulating a scenario as real as possible. Table 1 depicts the number of elements used in our environment and the percentages obtained for them.

In order to establish the population, we defined an initial population of 3,000 individuals, which is increased with other 3,000 individuals in each step. In Table 2 it is depicted the relationships between the individuals and the statements generated by the Reasoner, in order to show the complexity of our ontology. Table 2 shows that the number of statements is proportionally increased according to the number of individuals.

Figure 8 depicts the time in milliseconds (ms) used by the Reasoner to maintain updated the ontology with the network information and make decisions to change its configuration, considering the policies (*Reasoning time*) and the time needed to notify the decisions to the network applications in charge of switching on/off the base stations (*Query time*).

The last experiment shows that, having 9,750 devices and 150 base stations, our solution needs 570 ms to update the ontology with the network information and take the decision of changing the network configuration (or not). We consider that 570 ms is an acceptable time considering the number of active devices and base stations managed by our solution.

In this section, we have demonstrated that our solution can support a very large number of individuals or statements; when this number is linearly increased, the reasoning time also increases linearly. Furthermore, it is important to notice that the reasoning process could be made in an offline-mode depending on the scenario' necessities. Regarding the query time, we have demonstrated that queries do not have a significant impact in the performance of our solution.

Table 2 Individuals and statements per population

Population	1	2	3	4	5
Individuals	3,000	6,000	9,000	12,000	15,000
Statements	11,770	23,062	34,330	45,622	56,890

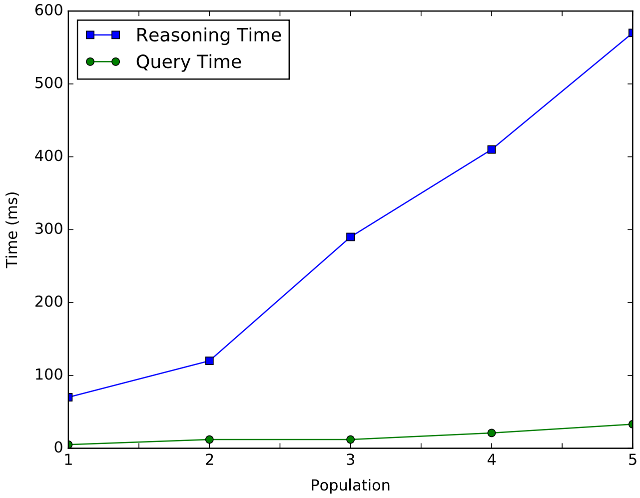


Fig. 8 Time needed by our solution to make the decision of modifying the network configuration

7 Conclusions and future work

A mobility-aware policy-based system oriented to the SDN paradigm has presented in this paper, with the aim of managing the energy efficiency of any network infrastructure at run-time by using policies. These policies allow the SDN paradigm to switch on/off network devices when they consume energy in an inefficient fashion, and also virtualize network resources such as proxies. This latter allows reducing traffic of specific services consumed by users who move close to the network infrastructure.

As future work, we plan to consider the location of devices and the users' behavior patterns in order to make decisions of switching or visualizing the network infrastructure. Having information about users' behavior allows our solution to predict users' movements and manage network resources in an energy saving fashion. In addition, we also plan the protection of the sensitive information. To this end, we could extend our solution to provide privacy-preserving policies, where users could indicate the granularity at which they want to release their location and behavior patterns.

Acknowledgments This work has been supported by a Séneca Foundation grant within the Human Resources Researching Training Program 2014, the European Commission Horizon 2020 Programme under grant agreement number H2020-ICT-2014-2/671672 - SELFNET (*Framework for Self-Organized Network Management in Virtualized and Software Defined Networks*), the Spanish MINECO (project DHARMA, *Dynamic Heterogeneous Threats Risk Management and Assessment*, with code TIN2014-59023-C2-1-R; and project

TecMASAS, *Techniques to Improve the Architecture of Servers, Applications and Services*, with code TIN2015-66972-C5-3-R), and the European Commission (FEDER/ERDF).

References

1. Horvath R, Nedbal D, Stieninger M (2015) A literature review on challenges and effects of software defined networking. Procedia Comput Sci 64:552–561
2. Huertas Celdrán A, Gil Pérez M, García Clemente FJ, Martínez Pérez G. Enabling highly dynamic mobile scenarios with software defined networking. IEEE Communications Magazine, Feature Topics Issue on SDN Use Cases for Service Provider Networks, In Press
3. Jimenez JM, Romero O, Rego A, Dilendra A, Lloret J (2015) Study of multimedia delivery over software defined networks. Netw Protoc Algorithm 7(4):37–62
4. Molina E, Jacob E, Astarloa A (2016) Using OpenFlow to control redundant paths in wireless networks. Netw Protoc Algorithm 8(1):90–103
5. Jingjin W, Yujing Z, Zukerman M, Yung EKN (2015) Energy-efficient base-stations sleep-mode techniques in green cellular networks A survey. IEEE Commun Surv Tutor 17(2):803–826
6. Auer G, Giannini V, Desset C, Godor I, Skillermark P, Olsson M, Imran MA, Sabella D, Gonzalez MJ, Blume O, Fehske A (2011) How much energy is needed to run a wireless network? IEEE Wirel Commun 18(5):40–49
7. Yun W, Staudinger J, Miller M (2012) High efficiency linear GaAs MMIC amplifier for wireless base station and Femto cell applications. In: IEEE Topical Conference on Power Amplifiers for Wireless and Radio Applications, pp 49–52
8. Marsan MA, Chiaraviglio L, Ciullo D, Meo M (2009) Optimal energy savings in cellular access networks. In: IEEE International Conference on Communications Workshops, pp 1–5
9. Claussen H, Ashraf I, Ho LTW (2010) Dynamic idle mode procedures for femtocells. Bell Labs Tech J 15(2):95–116
10. Rongpeng L, Zhifeng Z, Xianfu C, Palicot J, Honggang Z (2014) TACT: A transfer actor-critic learning framework for energy saving in cellular radio access networks. IEEE Trans Wirel Commun 13(4):2000–2011
11. Zhisheng N, Yiqun W, Jie G, Zexi Y (2010) Cell zooming for cost-efficient green cellular networks. IEEE Commun Mag 48(11):74–79
12. Bhaumik S, Narlikar G, Chattopadhyay S, Kanugovi S (2010) Breathe to stay cool: Adjusting cell sizes to reduce energy consumption. In: First ACM SIGCOMM Workshop on Green Networking, pp 41–46
13. Richter F, Fehske AJ, Fettweis GP (2009) Energy efficiency aspects of base station deployment strategies for cellular networks. In: IEEE Vehicular Technology Conference Fall, pp 1–5
14. Yulong Z, Jia Z, Rui Z (2013) Exploiting network cooperation in green wireless communication. IEEE Trans Commun 61(3):999–1010
15. Ming L, Pan L, Xiaoxia H, Yuguang F, Glisic S (2015) Energy consumption optimization for multihop cognitive cellular networks. IEEE Trans Mob Comput 14(2):358–372
16. Andrade S, Ruiz E, Granell E, Lloret J (2013) Energy consumption of wireless network access points. In: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 113, pp 81–91
17. Chang L, Jun Z, Letaief KB (2013) Energy efficiency analysis of small cell networks. In: International Conference on Communications, pp 4404–4408
18. Chia Y-K, Sun S, Zhang R (2013) Energy cooperation in cellular networks with renewable powered base stations. In: IEEE Wireless Communications and Networking Conference, pp 2542–2547
19. Kejiang Y, Dawei H, Xiaohong J, Huajun C, Shuang W (2010) Virtual machine based energy-efficient data center architecture for cloud computing: A performance perspective. In: Conference on Cyber, Physical and Social Computing, pp 171–178
20. Hyojoon K, Feamster N (2013) Improving network management with software defined networking. IEEE Commun Mag 51(2):114–119
21. Wang X, Vasilakos AV, Chen M, Liu Y, Kwon TT (2012) A survey of green mobile networks Opportunities and challenges. Mob Netw Appl 17(1):4–20
22. Huaping S, Kumar M, Das SK, Wang Z (2004) Energy-efficient caching and prefetching with data consistency in mobile distributed systems. In: International Parallel and Distributed Processing Symposium 2004, p 67
23. Aligrudic A, Pejanovic-Djurisic M (2014) Energy efficiency metrics for heterogenous wireless cellular networks. In: 2014 Wireless Telecommunications Symposium, pp 1–4
24. Motik B, Patel-Schneider PF, Parsia B (2012) OWL 2 web ontology language: Structural specification and functional-style syntax, 2nd edn. W3C Recommendation
25. Stanford Center for Biomedical Informatics Research. The Protégé tool: A free, open source ontology editor and knowledge-base framework. Available at http://protege.stanford.edu
26. Distributed Management Task Force, Inc. The CIM standard: Common Information Model. Available at http://www.dmtf.org/standards/cim
27. Linux Foundation. OpenDaylight: Open source SDN platform. Available at http://www.opendaylight.org
28. Horrocks I, Patel-Schneider PF, Boley H, Tabet S, Grosof B, Dean M (2004) SWRL: A semantic web rule language combining OWL and RuleML, W3C Member Submission
29. Sirin E, Parsia B, Cuenca Grau B, Kalyanpur A, Katz Y (2007) Pellet: A practical OWL-DL reasoner. Web Semant Sci Serv Agents World Wide Web 5(2):51–53
30. Prud'hommeaux E, Seaborne A (eds) (2008) SPARQL query language for RDF, W3C Recommendation