Switch-On/Off Policies for Energy Harvesting Small Cells through Distributed Q-Learning

Marco Miozzo[‡], Lorenza Giupponi[‡], Michele Rossi[†], Paolo Dini[‡]

[‡]*CTTC/CERCA, Av. Carl Friedrich Gauss, 7, 08860, Castelldefels, Barcelona, Spain*
[†]*DEI, University of Padova, Via G. Gradenigo, 6/B, 35131, Padova, Italy*
{mmiozzo, lgiupponi, pdini}@cttc.es, rossi@dei.unipd.it

***Abstract*—The massive deployment of small cells (SCs) represents one of the most promising solutions adopted by 5G cellular networks to meet the foreseen huge traffic demand. The high number of network elements entails a significant increase in the energy consumption. The usage of renewable energies for powering the small cells can help reduce the environmental impact of mobile networks in terms of energy consumption and also save on electric bills. In this paper, we consider a two-tier cellular network architecture where SCs can offload macro base stations and solely rely on energy harvesting and storage. In order to deal with the erratic nature of the energy arrival process, we exploit an ON/OFF switching algorithm, based on reinforcement learning, that autonomously learns energy income and traffic demand patterns. The algorithm is based on distributed multi-agent Q-learning for jointly optimizing the system performance and the self-sustainability of the SCs. We analyze the algorithm by assessing its convergence time, characterizing the obtained ON/OFF policies, and evaluating an offline trained variant. Simulation results demonstrate that our solution is able to increase the energy efficiency of the system with respect to simpler approaches. Moreover, the proposed method provides an harvested energy surplus, which can be used by mobile operators to offer ancillary services to the smart electricity grid.**

***Index Terms*—Mobile Networks, HetNet, SON, Sustainability, Renewable Energy, Energy Efficiency, Q-Learning.**

I. Introduction

Mobile networks will have to support a much higher capacity demand [1]. The fifth generation (5G) is expected to handle $1,000$ times more capacity per unit area with respect to 4G, especially within urban areas. The main drawback of this new technology is represented by its much increased energy consumption and on the consequent increase in greenhouse gasses (GHG) emissions. In fact, it is estimated that the worldwide ICT ecosystem could use as much as 51% of global electricity by 2030, which would translate in contributing to up to the 23% of the globally released GHG emissions [2]. This led 3GPP, mobile vendors and operators to put the energy sustainability in their roadmaps to 5G, extending the design on the next generation technologies accordingly.

Massive deployment of small form factor base stations (BSs), also referred to as small cells (SCs), represents the most promising architecture to meet the high capacity demands of mobile networks. Their reduced energy requirements encourages the use of renewable energy sources (RES) as distributed power suppliers. Their adoption is expected to have a twofold positive effect: 1) it will increase the use of renewable sources to provide energy, and consequently to reduce the carbon footprint of ICT, and 2) it will allow savings on power grid bills. Solar energy is probably the most important RES, due to its widespread availability, the good efficiency of photovoltaic (PV) technology and its competitive cost [3]. However, the resulting panel sizes may represent an obstacle for urban scenarios, where SCs are likely to be installed in street furnitures (i.e., traffic lamps, street lights, transportation hubs, etc.). Small form-factor solar panels can also be adopted by intelligently allocating energy to the SCs, putting them in power saving (OFF) mode when necessary, and exploiting the macro BS to compensate for their OFF time. The bottom line is that the panel size can be made small at the cost of some extra processing / optimization, which entails a tight interaction among SCs and between SCs and the macro BS.

Self Organized Networking (SON) aims at integrating intelligence and autonomous adaptability to network elements for improving the overall system efficiency and reducing complicated human operations for managing such a massive number of SCs. On this matter, researchers have been investigating sleep (ON/OFF) strategies to tackle the energy-efficiency (EE) problem with promising results [4]. However, with the introduction of energy harvesting (EH), we also need to consider the erratic and intermittent nature of RES, which further complicates the EE problem and the corresponding ON/OFF strategies. Most of the previous papers published in this area have only provided guidelines for dimensioning the network, while on-line approaches to control network elements have appeared only recently. In [5], the authors present an algorithm for determining when to switch OFF the SCs by solving a ski rental problem. The analysis is carried out considering Poisson arrivals for energy and traffic, which may provide a non-realistic approximation to these processes. A solution based on Reinforcement Learning (RL) is presented in [6], where the authors concentrate on the performance of a single SC. However, the impact of multiple SCs simultaneously switching OFFs within the same area is not considered.

In this paper, we fill these gaps by proposing a solution considering multiple SCs in a macro BS area, realistic traffic conditions and solar radiation data from real measurements. SC network is modeled as a multi-agent system where each agent (SC) makes autonomous decisions, according to a Decentralized SON (D-SON) paradigm. SCs supplied by solar energy and batteries (energy storage) are utilized as an overlay

layer in a two-tier network with a macro BS powered by the electricity grid. The behavior of small cells can be optimized to offload the traffic from the macro BS according to the energy income and the traffic demand. To this purpose, we designed a distributed on-line solution based on multi-agent RL, known as *distributed Q-learning*, which allows SCs to independently learn a radio resource management (RRM) policy. We preliminary tested the performance of this algorithm in [7]. This work extends our previous study by (i) investigating the convergence of the online algorithm, (ii) proposing an offline trained algorithm, (iii) characterizing the ON/OFF switching policies and (iv) calculating the surplus energy that cannot be stored, due to the energy storage capacity constraints. The results demonstrate that the proposed solution meets the design goals in terms of capacity and energy efficiency. Moreover, they spur new technical and business scenarios for mobile operators. In fact, SCs implementing our distributed Q-learning strategy may act as prosumers in a smart grid and use their surplus energy to, e.g., provide ancillary services to the electricity grid.

The remainder of the paper is organized as follows. In Section II we present the system model. Section III gives an overview on the distributed Q-learning algorithm, whereas the two proposed algorithms are presented in Section IV. In Section V we discuss some performance results. In Section VI we draw our conclusions and discuss future research directions.

II. System Model

We consider a two-tier network composed of M macro BSs and N SCs. The macro BSs are connected to the power grid and provide baseline coverage. The SCs are deployed in a hotspot manner to increase the system capacity, where needed (e.g., shopping hall, city center, etc.). SCs are solely powered through solar-harvested energy and possess rechargeable batteries to store the harvested energy.

For the BS power consumption model we use the linear model, $P = P_0 + \beta\rho$, where $\rho \in [0, 1]$ is the BS traffic load, normalized with respect to its maximum capacity, and P_0 is its baseline power consumption. This model is supported by real measurements [8] and closely matches the real power profile of BSs. The values of β and P_0 for the macro BS (SC) are 600 (39)W and 750 (105.6), respectively. We consider medium scale factor "metro cells" as SCs, featuring a maximum transmission power of 38 dBm.

The user equipment (UE) resource allocation scheme uses the methodology defined in [9]. This includes a detailed wireless channel model and the dynamic selection of the modulation and coding scheme (MCS) for each user as function of its channel state.

III. Distributed Q-Learning

Distributed Q-learning is an online optimization technique to control multi-agent systems, i.e., a system featuring N distributed agents (the SCs) which make decisions (switch-ON/OFF) in an uncoordinated fashion. Each agent has to independently learn a policy (switch-ON/OFF) through real-time interactions with the environment. These interactions entail taking actions for the agents and receiving, in return, a reward from the environment. In distributed Q-learning each agent i maintains a local policy and a local Q-function $Q(x_t^i, a_t^i)$ that only depends on its state x_t^i and actions a_t^i, with t being the decision epoch (time). The agents only have a partial view of the overall system and their local states may differ since traffic load and energy income may be unevenly distributed. In particular, the input of the switch-ON/OFF algorithm depends on the SC location and on the geographical distribution of its users, affecting for instance, the experienced traffic load. The decision making process of each agent is defined according to a Markov Decision Process (MDP) with state vector $\boldsymbol{x}_t = (x_t^1, x_t^2, \ldots, x_t^N)$, where x_t^i is the state associated with SC i at time t. Agent i *independently* chooses an action a_t^i from an action set $\mathcal{A}$ based on its own state x_t^i. At the next decision epoch $t+1$, the agent receives a reward r_t^i from the environment. The *agent dependent* reward r_t^i is then used to locally update the Q-value, $Q(x_t^i, a_t^i)$, indicating the level of convenience of selecting action a_t^i when in state x_t^i. The Q-value is updated as follows:

$$\begin{aligned} Q(x_t^i, a_t^i) \leftarrow\ & Q(x_t^i, a_t^i) + \\ +\ & \alpha(x_t^i, a_t^i)[r_t^i + \gamma \max_a (Q(x_{t+1}^i, a) - Q(x_t^i, a))] \end{aligned} \tag{1}$$

where α is the learning rate, γ is the discount factor and x_{t+1}^i is the next state for agent i. α represents the speed of convergence. The *asynchronous Q-learning algorithm* proposed in [10], uses a learning rate given by a polynomial function that at time t accounts for the number of visits, up to and including time t, to state-action pair (x, a), termed $n(x, a, t)$. In detail $\alpha^\omega(x, a) = \alpha / n(x, a, t)^\omega$, where $\omega = 1$ leads to a linear learning rate, $\omega \in (1/2, 1)$ to a polynomial one and $\omega = 0$ to a constant learning rate.

To make the best decisions (exploitation) the algorithm must have gathered enough information from the environment (exploration). The exploration phase is commonly controlled by an *ε-greedy* approach, in which random states are visited by the agents with probability ε. Since rigorous convergence results for multi-agent reinforcement learning algorithms are still an open research question, here we refer to the convergence time as the first instant in which the Q-values remain stable within a certain tolerance. In particular, we say that the system has reached convergence when all the SC batteries are below B_{th} for a certain amount of time (e.g., within a window of consecutive days). The rationale behind this definition is to foster the energy sustainability of the SCs. For more details on RL and Q-learning the reader is referred to, e.g., [11].

IV. Algorithms

A. ON/OFF switching through online distributed Q-learning

At time t, the local state x_t^i of agent i is $x_t^i = (S_t^i, B_t^i, L_t^i)$, where S_t^i represents the amount of energy harvested (RES), B_t^i is the normalized battery energy level, L_t^i is the normalized load for SC i, which depends on the number of users, their position and their traffic demand. S_t^i, B_t^i and L_t^i have been uniformly quantized into 2, 5 and 3 levels, respectively. The set of possible actions $\mathcal{A}$ has been limited to switching ON and

OFF the SC, since the β parameter in the power consumption equation of Section II only marginally impacts the energy consumption of the SC. Also, with T_t^i we indicate the normalized throughput of SC i in slot t, D_t is the instantaneous *system* drop rate, defined as the ratio between the total amount of traffic dropped and the traffic demand in the entire network (accounting for macro and SCs) in slot t, whereas D_{th} is the maximum tolerable drop rate. Finally, B_{th} is a threshold on the battery level. The reward function is defined as:

$$r_t^i = \begin{cases} 0 & B_t^i < B_{\text{th}} \text{ or } D_t > D_{\text{th}} \\ \kappa T_t^i & B_t^i \geq B_{\text{th}} \text{ and } D_t \leq D_{\text{th}} \text{ and SC } i \text{ is ON} \\ 1/B_t^i & B_t^i \geq B_{\text{th}} \text{ and } D_t \leq D_{\text{th}} \text{ and SC } i \text{ is OFF} \end{cases} \quad (2)$$

The reward r_t^i is designed to avoid critical conditions such as low battery level or too high system drop rates, forcing the SC to turn OFF in such undesirable cases. In normal load conditions (i.e., $D_t \leq D_{\text{th}}$), the self-sustainability is incentivized by putting a reward proportional to the inverse of the energy buffer level ($1/B_t^i$) when the SC is OFF. On the other hand, in case the SC is ON the reward is proportional to the throughput, as this promotes offloading the macro BS. Note that the SC will be incentivized to switch ON when the reward coming from the throughput overweights that from energy saving (i.e., when $\kappa T_t^i > 1/B_t^i$). The reader is referred to [7] for additional details on the algorithm.

B. ON/OFF switching based on trained distributed Q-learning

Online learning algorithms suffer from an initial exploration phase to gather information from the environment and, based on this acquired knowledge, make good decisions. This process produces instability and poor performance potentially for a long time, i.e., until a sufficient amount of knowledge is gathered. We proposed an offline training period for the algorithm to setup initial switch OFF/ON policies. In detail, the TRAINING phase consists of running the agent with the energy statistics of a specific month for generating the Q-tables in an offline fashion. This returns the trained Q-values that can be used for initializing the Q-tables of the SCs when they are deployed in their ONLINE operative mode. The pseudocode of this solution is presented in Alg. 1. This initial training helps reduce the initial exploration phase and, in case the algorithm is not able to follow the dynamics of the environment, it also helps improve the system performance, by avoiding slow recalibration phases. We note that, the training phase can be either performed with a simulation approach, as we propose, or obtained by other *expert* SCs that have been already deployed, as in the *transfer learning* paradigm [12].

Algorithm 1 Trained Distributed Q-learning

procedure TRAINING($Q^m_{\text{init}}(x^i, a^i)$)
  $Q^m_{\text{init}}(x^i, a^i) \leftarrow 0$
  for $m \in \mathcal{M}$ **do**
    Run Q-learning(EH_m)
    $Q^m_{\text{init}}(x^i, a^i) \leftarrow Q\text{-table}(x^i, a^i)$
  end for
end procedure

procedure ONLINE
  for $m \in \mathcal{M}$ **do**
    for $m \in \mathcal{D}^m$ **do**
      $Q\text{-table}(x^i, a^i) \leftarrow Q^m_{\text{init}}(x^i, a^i)$
      Run Q-learning(EH_m)
    end for
  end for
end procedure

where
$\mathcal{M}$ is the set on months
EH_m = Energy Traces for month m
$\mathcal{D}^m$ is the set of days in month m

V. PERFORMANCE EVALUATION

A. Simulation Scenario

We consider a deployment of a varying number of SCs within a square macro cell area with a side of 1 km. The macro BS is placed in the center of it, whereas the SCs are randomly positioned with the constraint that their cells do not overlap. This translates into a minimum inter-SC distance of 50 m, which corresponds to the coverage radius of a SC with transmission power of 38 dBm. The coverage area of each SC is populated with 120 uniformly placed UEs, which allow congesting the SC in peak traffic hours. Data load is modeled using a urban profile [13], where traffic is concentrated around working hours and has one peak in the morning and one in the afternoon. According to [8], we considered that 20% of the UEs are "heavy users" with a data volume of 900 MB/h, while the remaining UEs are "ordinary users" (112.5 MB/h). As for the RES system, we consider the Panasonic N235B solar modules, which have single cell efficiencies of about 21%, delivering about 186 W/m^2. Each SC is equipped with an array of 16×16 solar cells (i.e., 4.48 m^2). The battery size is 2 kWh (panel and battery sizes have been chosen so that SC batteries can be replenished in a full winter day). Realistic harvested energy traces are obtained using the SolarStat tool [14], considering the city of Los Angeles as the deployment location.

The analysis is performed as follows. We first elaborate on the convergence of the online algorithm, then we characterize the switch ON/OFF policies in different representative months (i.e., January, April and July) and compare the performance of the *online* distributed Q-learning ("QL" in the figures) against that of a distributed Q-learning algorithm trained in an *offline* fashion ("QLT"). We conclude our investigation with an assessment of the energy efficiency of the considered techniques. QL and QLT are contrasted with a greedy scheme ("greedy" in the figures) where the $i-th$ SC is switched OFF at time t when its battery level B_t^i drops below B_{th}, and is reactivated at time $t+\Delta$ when it has harvested enough energy for returning above the threshold (i.e., $B_{t+\Delta}^i \geq B_{\text{th}}$). The battery threshold B_{th} is set to 20% of the battery capacity in

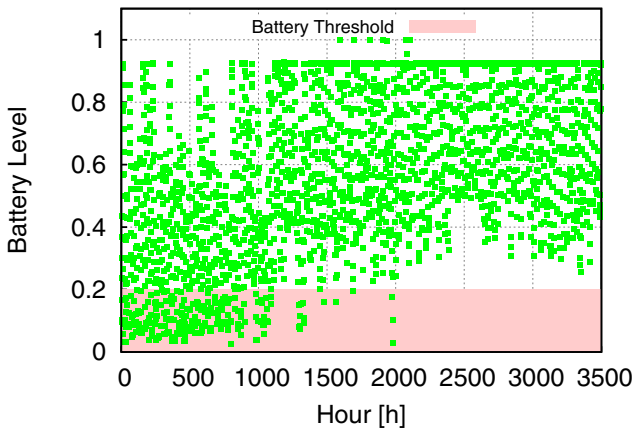

Fig. 1. Battery level for the month of January of a single SC.

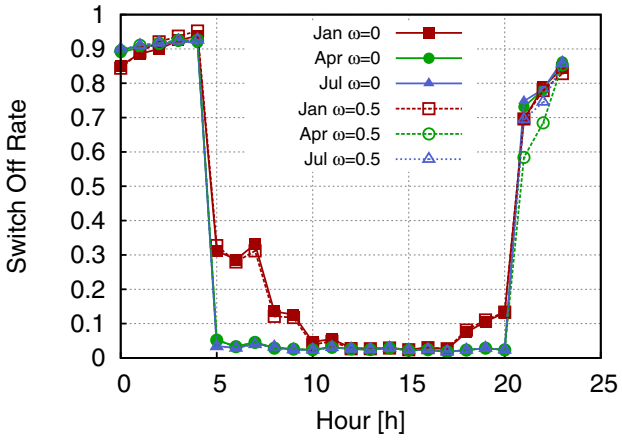

Fig. 2. Switch OFF rate of a SC during the day with a single SC.

order to keep the battery within its safe operating regime [6]. The threshold on the instantaneous traffic drop rate is set to $D_{\text{th}} = 0.05$, while the ε parameter of the ε-*greedy* exploration is set to 0.1.

B. Online Algorithm Convergence

At time t, a SC i is said to be in outage if $B_t^i \leq B_{\text{th}}$. Then, the total outage time for SC i over a period of time $T > 0$ is computed as $\int_0^T 1\{B_t^i \leq B_{\text{th}}\}dt$, where $1\{\cdot\}$ is the indicator function, which is one if the event in its argument is verified and zero otherwise. In a certain day, the system is said to be in outage if the total outage time, obtained summing the outage time of all the SCs during the day, is higher than 5%. An algorithm is said to have converged when no outage occurs during a window of three consecutive days. An example of the convergence behavior is shown in Fig. 1, where the hourly battery level of a SC is plotted on a per hour basis for the month of January. A preliminary phase of instability can be noted until hour 1000 (i.e., lasting about 40 days), where the SC adopts a greedy-like approach and drops frequently below the threshold since it is using the energy only according to instantaneous availability. After this amount of time, the agent has been able to gather information from the environment in order for its Q functions to stabilize. After that point, the battery level drops below B_{th} less often and the density of points starts becoming more prominent above the battery threshold. In proximity of 1300 hours, we can appreciate a temporary instability due to the scarce amount of energy harvested during several consecutive days. However, we note that the algorithm promptly reacts and drives the system toward a good (zero-outage) region. Similar considerations hold for scenarios involving multiple SCs.

C. Policy Analysis

The switch OFF rate of a single SC during 24 hours is reported in Fig. 2 for polynomial ($\omega = 0.5$) and constant ($\omega = 0$) learning rates for the months of January, April and July. The rate is calculated by simulating 180 days, so as to allow for the completion of the training phase and increase the statistical confidence of the results.

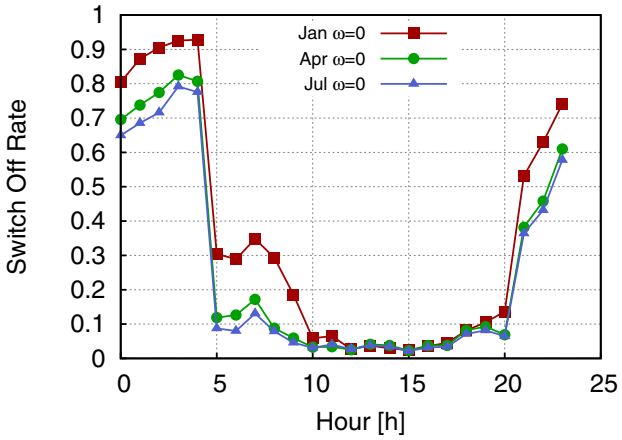

Fig. 3. Switch OFF rate of a SC during the day with multiple SCs.

Switch OFFs are more intensive during early morning hours (from 0am to 4am) and in the night (from 9pm to 11pm), due to the scarce harvested energy and the low traffic demand at night-time. When the SC is turned OFF, the SC agent chooses to recharge its battery and relies on the macro cell for serving the UEs within its coverage. This behavior is similar for all months. In January, another less intensive switch OFF period can be appreciated from 5am to 9am (switch OFF rate of about 0.3). This is due to a feeble harvesting process during those months. Moreover, it can be noticed that the two values of ω do not significantly affect the shape of the policy. This implies that a constant learning rate ($\omega = 0$) provides the needed flexibility for Q-learning to effectively cope with the system dynamics.

In Fig. 3, the switch OFF rate of a SC in a multi-cell scenario with 10 SCs is presented. The policies are similar to those in Fig. 2 (single cell case). However, we can appreciate a slight reduction in the switch OFF intensity in the early morning. Here, the SCs switch OFF less often in order not to overload the macro cell and maintain the traffic drop rate below D_{th}. In the months of April and July, the number of switch OFFs is lower than in January, due to an increase in the harvested energy income. According to this, we can appreciate

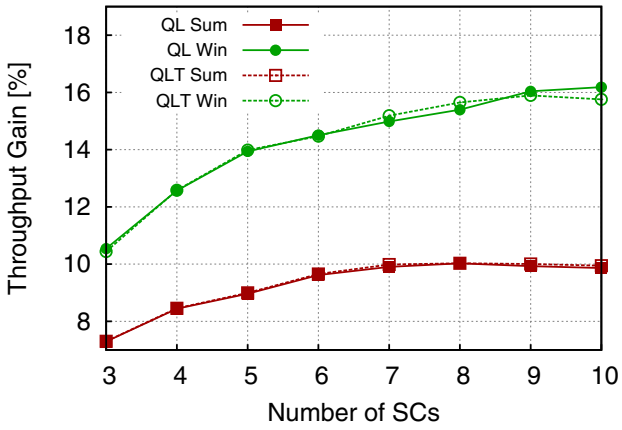

Fig. 4. Average throughput gain [%] of QL and QLT with respect to the greedy scheme.

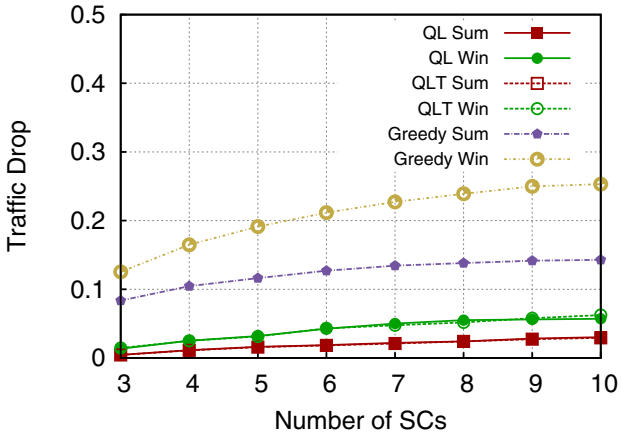

Fig. 5. Traffic drop rate for QL, QLT and greedy.

that the algorithm is able both to learn the policy as function of the energy and traffic patterns and to adapt to different scenarios with different number of SCs.

D. Network Performance

In Fig. 4 we show the average throughput gain of QL and QLT with respect to the greedy scheme by varying the number of SCs, whereas in Fig. 5 we show the traffic drop rate of QL, QLT and of the greedy algorithm. The results are achieved running simulations across a full year. Statistics are gathered only when the algorithm has converged and for a duration of 365 days. Since the harvesting process substantially differs for different seasons, we have presented our results separately for the *winter* and the *summer* periods, respectively termed "Win" and "Sum" in the plots. January, February, October, November and December are considered *winter* months.

The effect of ω is not relevant from the throughput and traffic drop perspective. QL and QLT outperform greedy: the throughput gain of Q-learning with respect to greedy is of up to 16% in the winter, which results in a drop rate smaller than 5% for QL and QLT, whereas the drop rate reaches 20% for the greedy scheme. The difference between winter and summer resides in the corresponding switch OFF policies, as discussed in the previous section.

We also note that QL and QLT have similar performance, both in terms of throughput and traffic drop, but they have a different convergence time. In fact, QLT presents 6 times shorter convergence times on average, taking at most 10 days to converge in the worst case scenario of 10 SCs, with respect to the 40 days needed by QL in the same settings. However, QL can rapidly adapt to the changing dynamics of the harvesting process across the months (as reported in Fig. 4 and Fig. 5), thus rendering useless the per-month-training of QLT. Therefore, SC agents shall only be trained to gather the necessary information during their initial exploration phase. Upon that, they can be used in an online fashion and further training is no longer required.

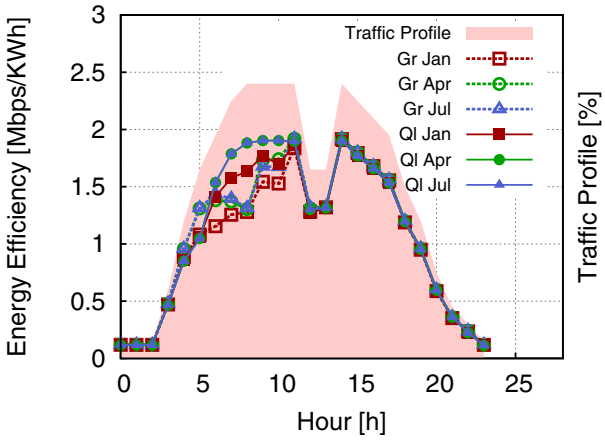

Fig. 6. Average energy efficiency of a SC during the day with a single SC.

E. Energy Efficiency

In this section, the energy performance of QLT is not shown as is the same as that of QL. The energy efficiency is defined as $\text{EE} = T_S/E_S$, where T_S is the system throughput and E_S is the total energy drained by the macro BS (from the power grid). The traffic demand profile is also shown.

The energy consumption metric is shown in Fig. 6, where the QL energy efficiency is compared with that of the greedy scheme for January, April and July. QL outperforms the greedy scheme during the morning slot (e.g., from 6am to 12pm), since it saves energy during the nocturnal low traffic period in order to have enough energy reserve for the morning peaks of traffic, without compromising the throughput performance.

Fig. 7 reports the energy efficiency improvement of QL with respect to greedy, varying the number of SCs. QL offers a considerable gain, which reaches 15% in the winter months. This is due to its higher throughput, which follows from a proper usage of the available energy reserves. The lower gain during the summer months and its decreasing behavior for an increasing number of SCs are motivated by the fact that the RES system has been dimensioned to provide the necessary energy in the worst case, which is a winter day. This implies that, during the summer, there are days in which the algorithm

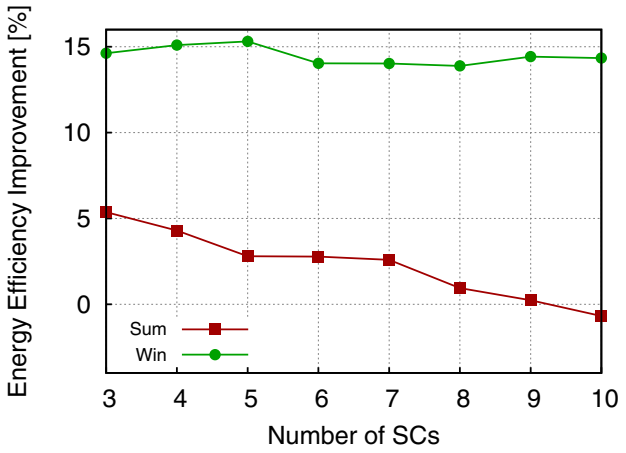

Fig. 7. Energy efficiency improvement [%] of QL with respect to greedy vs number of SCs.

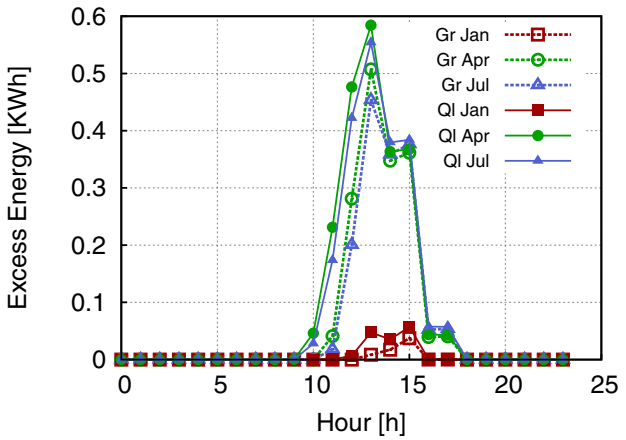

Fig. 8. Average redundant energy during the day for a single SC.

does not have to smartly save energy, since the harvested energy is enough for the whole day and therefore greedy and and QL have similar performance. In this case, the abundant energy has to be discarded by the SCs, i.e., it can neither be used for transmission nor stored in the battery. This fact is shown in Fig. 8 ("excess energy").

With QL, the total amount of grid energy drained by the system spans from 7.3 KWh for a network of 3 SCs, to 7.9 KWh with 10 SCs (as compared to 7.5 KWh for the greedy scheme). Note that QL has a worse annual energy consumption performance since it serves more traffic, as we have discussed in Section V-D. QL has a higher energy surplus than greedy. In fact, QL (greedy) reserves 0.124 (0.064) KWh in January, 2.115 (1.619) KWh in April and 2.021 (1.513) KWh in July. This translates into a total amount of energy not used by QL of 400 KWh in the summer (300 for greedy), and of 65 KWh in the winter (36 for the greedy scheme). Considering the higher energy efficiency and the energy surplus of QL, we conclude that QL uses less energy to offload the macro BS. In such a context, SCs may act as prosumers and offer/trade their excess energy to provide ancillary services to the smart grid.

VI. Conclusions

In this paper, we have presented a distributed implementation of a switch OFF/ON algorithm aimed at optimizing the energy usage in a dense small cell deployment with solar energy harvesting capability. The solution uses Q-learning techniques to learn the dynamics of energy harvesting and traffic processes and make switch ON/OFF decisions accordingly. Our numerical results demonstrate that distributed learning is a promising approach to make decisions in complex, dense and dynamic scenarios, leading to substantial advantages with respect to greedy schemes, such as higher throughput and energy efficiency.

Acknowledgment

The research leading to these results has received funding by the Spanish Ministry of Economy and Competitiveness under grant TEC2014-60491-R (Project 5GNORM) and the European Unions Horizon 2020 research and innovation programme under SANSA (H2020-645047) grant.

References

[1] Cisco Systems Inc., "Cisco visual networking index global mobile data traffic forecast update 2013-2018," White Paper, http://www.cisco.com/, Feb. 2013.

[2] A. S. G. Andrae and T. Edler, "On global electricity usage of communication technology: Trends to 2030," *Challenges*, vol. 6, no. 1, p. 117, 2015.

[3] G. Piro, M. Miozzo, G. Forte, N. Baldo, L. A. Grieco, G. Boggia, and P. Dini, "HetNets Powered by Renewable Energy Sources," *IEEE Internet Computing*, vol. 17, no. 1, pp. 32–39, 2013.

[4] H. Al Haj Hassan, L. Nuaymi, and A. Pelov, "Renewable energy in cellular networks: A survey," in *IEEE Online Conference on Green Communications (GreenCom)*, Oct. 2013.

[5] G. Lee, W. Saad, M. Bennis, A. Mehbodniya, and F. Adachi, "Online ski rental for scheduling self-powered, energy harvesting small base stations," in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6.

[6] M. Mendil, A. D. Domenico, V. Heiries, R. Caire, and N. Hadjsaid, "Fuzzy Q-Learning based Energy Management of Small Cells Powered by the Smart Grid," in *IEEE PIMRC*, Valencia, Spain, 2016.

[7] M. Miozzo, L. Giupponi, M. Rossi, and P. Dini, "Distributed Q-Learning for Energy Harvesting Heterogeneous Networks," in *IEEE ICC 2015 workshop on Green Communications and Networks with Energy Harvesting, Smart Grids and Renewable Energies*, London, United Kingdom, 2015.

[8] EU EARTH: Energy Aware Radio and neTwork tecHnologies, "D2.3: Energy efficiency analysis of the reference systems, areas of improvements and target breakdown," Deliverable D2.3, www.ict-earth.eu, 2010.

[9] M. Mezzavilla, M. Miozzo, M. Rossi, N. Baldo, and M. Zorzi, "A Lightweight and Accurate Link Abstraction Model for System-Level Simulation of LTE Networks in ns-3," in *ACM MSWIM*, Paphos, Cyprus Island, Oct. 2012.

[10] E. Even-Dar and Y. Mansour, "Learning rates for q-learning," *J. Mach. Learn. Res.*, vol. 5, pp. 1–25, Dec. 2004.

[11] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[12] L. Giupponi, A. M. Galindo-Serrano, and M. Dohler, "From cognition to docition: The teaching radio paradigm for distributed & autonomous deployments," *Comput. Commun.*, vol. 33, no. 17, pp. 2015–2020, Nov. 2010.

[13] M. Marsan, G. Bucalo, A. Di Caro, M. Meo, and Y. Zhang, "Towards zero grid electricity networking: Powering BSs with renewable energy sources," in *IEEE International Conference on Communications (ICC)*, Budapest, Hungary, Jun. 2013.

[14] M. Miozzo, D. Zordan, P. Dini, and M. Rossi, "SolarStat: Modeling Photovoltaic Sources through Stochastic Markov Processes," in *IEEE Energy Conference (ENERGYCON)*, Dubrovnik, Croatia, May 2014.