



Balancing the use of batteries and opportunistic scheduling policies for maximizing renewable energy consumption in a Cloud data center

Yunbo Li, Anne-Cécile Orgerie, Jean-Marc Menaud

► To cite this version:

Yunbo Li, Anne-Cécile Orgerie, Jean-Marc Menaud. Balancing the use of batteries and opportunistic scheduling policies for maximizing renewable energy consumption in a Cloud data center. PDP 2017: 25th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, Mar 2017, St Petersburg, Russia. 10.1109/PDP.2017.24 . hal-01432752

HAL Id: hal-01432752

<https://hal.inria.fr/hal-01432752>

Submitted on 12 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Balancing the use of batteries and opportunistic scheduling policies for maximizing renewable energy consumption in a Cloud data center

Yunbo Li
Ecole des Mines de Nantes, LINA
IRISA, Rennes, France
Email: yunbo.li@mine-nantes.fr

Anne-Cécile Orgerie
CNRS - IRISA, Rennes, France
Email: anne-cecile.orerie@irisa.fr

Jean-Marc Menaud
Ecole des Mines de Nantes, LINA
Nantes, France
Email: menaud@mine-nantes.fr

Abstract—The fast growth of cloud computing considerably increases the energy consumption of cloud infrastructures, especially, data centers. To reduce brown energy consumption and carbon footprint, renewable energy such as solar/wind energy is considered recently to supply new green data centers. As renewable energy is intermittent and fluctuates from time to time, this paper considers two fundamental approaches for improving the usage of renewable energy in a small/medium-sized data center. One approach is based on opportunistic scheduling: more jobs are performed when renewable energy is available. The other approach relies on Energy Storage Devices (ESDs), which store renewable energy surplus at first and then, provide energy to the data center when renewable energy becomes unavailable. In this paper, we explore these two means to maximize the utilization of on-site renewable energy for small data centers. By using real-world job workload and solar energy traces, our experimental results show the energy consumption with varying battery size and solar panel dimensions for opportunistic scheduling or ESD-only solution. The results also demonstrate that opportunistic scheduling can reduce the demand for ESD capacity. Finally, we find an intermediate solution mixing both approaches in order to achieve a balance in all aspects, implying minimizing the renewable energy losses. It also saves brown energy consumption by up to 33% compared to ESD-only solution.

Keywords—Green data centers; Cloud computing; renewable energy; energy storage; opportunistic scheduling.

I. INTRODUCTION

The fast growing demand for Cloud services, emphasized by the data deluge, has lead to a boom in the number of data centers worldwide. Consequently, the global electricity part dedicated to their consumption has reached unprecedented levels. In 2012, the number of data centers worldwide was estimated at 509,147 consuming roughly the output of 30 nuclear power plants [1]. In 2016, another study estimates that worldwide the data centers use 91 billion kilowatt-hours of electricity – enough to power New York City twice over – and their consumption is still growing rapidly [2]. This situation raises major environmental, economic and social concerns.

One way to save energy at a data center level consists in locating it close to where the electricity is generated, hence minimizing transmission losses. For example, Western North Carolina, USA, attracts data centers with its low electricity prices due to abundant capacity of coal and nuclear power

following the departure of the region’s textile and furniture manufacturing [3]. Currently, this region has three super-size data centers from Google, Apple and Facebook with respective power demands of 60 to 100 MW, 100 MW and 40 MW [3]. However, such huge facilities represent only a small fraction of the global consumption of data centers. Indeed, small- and medium-sized server rooms continue to account for nearly half the electricity consumption of the market [4].

For economical, environmental or marketing reasons, other companies opt for greener sources of energy. For example, Quincy (Washington, USA) supplies electricity to data facilities from Yahoo, Microsoft, Dell and Amazon with its low-cost hydro-electrics left behind following the shutting down of the region’s aluminum industry [3]. Several renewable energy sources like wind power, solar energy, hydro-power, bio-energy, geothermal power and marine power can be considered to power up super-sized facilities. The production variability of most renewable sources leads data center facilities to only partially rely on them and to depend also on the regular electrical grid as a backup or main supplier.

While using renewable sources may reduce energy costs, reduce peak power costs, or both [5], they are mostly intermittent and fluctuating over time (like sun and wind for instance). These variations may lead to electricity losses if the computing workload does not match the renewable production. Cloud infrastructures, on the other hand, can take advantage of multiple locations to increase their green consumption with approaches such as follow-the-sun and follow-the-wind [6]. As sun and wind provide renewable sources of energy whose capacity fluctuates over time, the rationale is to place computing jobs on resources using renewable energy, and migrate jobs as renewable energy becomes available for resources in other locations.

In the case of a single data center, such follow-the-sun approaches are not feasible. But instead, opportunistic scheduling algorithms can make advantage of renewable energy availability to perform jobs with low priorities [7]. Opportunistic policies distinguish two kind of computing jobs: jobs requiring to run continuously (like web servers) and jobs that can be delayed and interrupted, but with a deadline constraint (such batch jobs include monthly payroll computation for example).

The jobs of the second type wait for renewable energy surplus to be scheduled, thus reducing the overall consumption part of brown energy. However, such scheduling policies make use of virtual machine migrations and suspend/resume functions that have a cost in terms of energy consumption [8].

Another possible method for improving the effective utilization of intermittent and fluctuating renewable energy consists in using batteries to store green production surplus, and to use it during low production periods [5]. Typically for solar sources, energy can be stored during the day – if not fully consumed – and be utilized during nights when there is no production. However, batteries have an inherent energy efficiency (their yield) that leads to energy losses. So, is it greener to use opportunistic scheduling or batteries?

In this paper, we discuss these two approaches for maximizing the utilization of renewable energy in small and medium data centers. We compare these two solutions in terms of renewable energy utilization and total energy consumption in order to estimate whether the losses due to the battery efficiency balances or not the losses due to migration costs incurred by opportunistic scheduling policies. We also evaluate an intermediate solution mixing both approaches. This study investigates two types of batteries (lead-acid and lithium-ion, but can be easily generalized to other types of ESD), the optimal size of photovoltaic panels, several sunlight profiles and real-world workload traces from a medium-sized data center.

The remainder of the paper is organized as follows. Section II presents recent work on reducing energy consumption in data centers. Section III describes our data center and Energy Storage Device (ESD) models. Section IV describes the job scheduling algorithms: baseline algorithm with ESD, opportunistic scheduling without ESD and opportunistic scheduling combined with ESD. Section V presents our experimental setup including an analysis of real-world workload traces and the simulation-based methodology to find the optimal solar panel dimension and battery size for a given data center and a given workload. Section VI presents the results of our simulations which show the relationship between brown energy consumption and different solar panel dimension/battery size. Lastly, Section VII concludes this work and discusses the challenges for future research.

II. BACKGROUND AND RELATED WORK

Data center consumption reduction. Much of the early work focuses on the reduction of the brown energy consumption in data centers [5], [9]–[11]. In order to save energy in mono-site data center, a classical goal consists in reducing the number of powered-on servers as idle servers can typically consume half of their peak power consumption [8]. Job scheduling, virtual machines (VMs) placement and consolidation directly affect the number of powered-on servers. In [12], Chaima et al. propose an energy-aware allocation and consolidation algorithm to switch off the under-utilized servers in cloud data centers. For powered-on servers, dynamic voltage and frequency scaling (DVFS) allows CPU

to work at multiple frequencies by increasing/decreasing voltage to meet the performance demand and to reduce the average power consumption [13]–[15]. Lastly, resource over-commitment increases server utilization and minimizes the number of powered-on servers. In [16], Zhang et al. design a migration algorithm that reduces the risk of overload and minimizes the number of VM migrations in over-committed data centers.

Integrate renewable energy into data center. Unlike traditional infrastructures where energy sources are controllable, integrating renewable energy into a data center is difficult due to its intermittent and variable nature. Usually, solar energy is considered as an admissible renewable source as solar panels are easy to install, they present a reasonable efficiency and the variations in their electricity production are not too abrupt (as for wind) [5]. In [17], Andrew et al. point out that the slack time is a key feature enabling the jobs to be delayed until the renewable energy becomes available. They propose an off-line scheduling algorithm to align the brown energy consumption with the renewable energy supply. However, it is arduous to make accurate long-range workload prediction. In the real-world workloads, a number of jobs cannot be delayed or interrupted, such as web services for instance. In our previous work [7], we propose an on-line solution making use of opportunistic scheduling for increasing solar energy utilization in a small-/medium-size data center without energy storage. This approach leverages two ideas: 1) delay part of the jobs (interruptible ones) until solar energy becomes available; 2) when the renewable energy production cannot fully support the entire workload energy consumption, the system switches off the under-utilized servers with the help of consolidation techniques. However, it resorts to a consequent number of migrations caused by the consolidation algorithm and thus incurs a non negligible energy overhead. Furthermore, it may waste some renewable energy if there is not enough work to consume it all during production peaks. Batteries can solve this issue by partially or fully storing renewable energy surplus.

Energy Storage Devices. The variable and intermittent nature of renewable energy – like solar energy – makes it difficult to manage. In order to increase the usage of renewable energy, one way consists in carefully scheduling the workload to align with the time-varying available renewable energy. An alternate solution consists in using ESDs [18] to store the renewable energy and generate electricity for later usage.

The main parameters to be considered when dealing with ESDs are:

- 1) Efficiency: The energy used to charge a battery is higher than what can be used at a later time.
- 2) Battery charging and Discharging Rate Limit: This charging/discharging rate limit is determined by the type of battery. Typically the discharge/charge ratio is larger than 1 for most batteries.
- 3) Self-Discharge: There is an energy loss which is proportional to the storage time.
- 4) Depth-of-Discharge (DoD): Many factors may impact the battery lifetime such as the charging/discharging

cycles [19], [20]. DoD can also impact the battery lifetime: in order to extend the battery lifetime to a reasonable time, we cannot use the full capacity of battery.

We now present the ESD that includes re-chargeable batteries technologies (Electrochemical). In this paper, we consider two kinds of batteries: Lead-Acid Battery (LA) and Lithium-Ion (LI) which are prevalent in current data centers. Table I shows the different constraints per battery kind.

	LA	LI
DoD	0.8	0.8
Charge rate / ESD size (%)	12.5	25
Efficiency	0.75	0.85
Self-discharge (per day)	0.3%	0.1%
Discharge rate / charge rate ratio	10	5
Price (\$/kWh)	200	525

TABLE I: The battery characteristics (data from [19]–[21])

In comparison with LA, LI battery has higher energy density, energy efficiency and lower discharge rate, but also higher cost. In the rest of the paper, battery and ESD terms are interchangeable.

III. CONTEXT AND ASSUMPTIONS

This paper is focusing on maximizing the use of renewable energy in a small/medium-scale data center with on-site solar panels. This section describes the context of this work.

A. Small and medium data centers

The considered data center comprises between 20 to 150 servers. Each server has limited resources in terms of CPU and RAM. We assume that there is no centralized storage system in the data center: each server has its own hard disk [22]. The data center is equipped with photovoltaic (PV) panels and an ESD. It has dual brown (from regular grid) and renewable energy supplies. If the renewable energy cannot be entirely consumed by the data center, the ESD stores the surplus of renewable energy for future use. We also assume that each server has a switch connected with renewable and brown energy supplies and the ESD. Specifically, the server can only opt for using one of the three sources at the same time.

A job can be submitted to the data center at anytime and it consists of an individual Virtual Machine (VM) to execute for a given duration. A VM is considered as the basic unit of resource allocation. We assume each VM has two constraints, namely CPU and RAM, and each job has its own duration and a predefined deadline. When a job finishes, the VM is destroyed and it releases its reserved resources back to the server. The job management system assumes that time is divided into slots. The VM resource allocation operations are performed periodically at the beginning of each time slot.

B. ESD model

As shown in Figure 1, PV panels turn solar energy into electrical energy which can be directly supplied for the data center or collected by the ESD. The ESD is composed of

rechargeable batteries which first collect and store energy (generated from solar energy only), and then power the data center when scheduled.

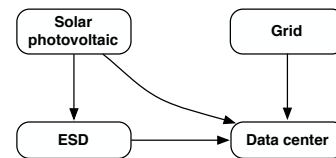


Fig. 1: Energy sources of the data center

The capacity of the ESD is finite. Herein, we use parameter C to express the maximum capacity of an ESD. At a given time t , $C_{available}(t)$ represents the energy that has been collected and is stored by the ESD. In order to keep a longer battery lifetime, we take into account the DoD constraint [20], which stipulates that the remaining energy stored in an ESD has to be larger than the DoD threshold. So, in other terms, the available stored energy is smaller than a higher bound ηC ($0 < \eta < 1$, e.g. $\eta = 0.8$). By considering the DoD constraint, one can see that $C_{available}(t)$ never reaches C . Formally, we have $0 \leq C_{available}(t) \leq \eta C$.

An ESD has two significant functionalities: charging (collects energy from solar panels) and discharging (powers the data center). In our system, we consider that charging and discharging are two independent procedures. It implies ESD is never under charging and discharging states simultaneously. The charging rate has an upper bound λ depending on the ESD type and capacity. During a given time period $[t_i, t_j]$ ($t_j > t_i$), if we suppose the available green energy (supplied by PV cells) is $E(t_i, t_j)$, we use formula 1 to compute the amount of energy $E_{in}(t_i, t_j)$ that can be collected by an ESD.

$$E_{in}(t_i, t_j) = \min(E(t_i, t_j), \lambda(t_j - t_i), C_{available}(t_i)) \times \sigma \quad (1)$$

Parameter σ is constant and expresses the energy efficiency of the battery's charging procedure. The discharging rate also has an upper bound denoted μ . During a consecutive time period $[t_i, t_j]$, we use formula ?? to compute the amount of energy $E_{out}(t_i, t_j)$ that can be provided by the ESD. Parameter $E_{self-discharge}(t_j - t_i)$ represents the energy loss due to the self-discharging of batteries.

$$E_{out}(t_i, t_j) = \min(\mu(t_j - t_i), \eta C - C_{available}) - E_{self-discharge}(t_j - t_i)$$

In this paper, we consider only solar energy as renewable energy source. Due to the variable and intermittent nature of solar energy, an energy production prediction is performed when a job scheduling decision has to be taken. It only predicts the solar energy for the following time slot (1 hour), so that such short-time prediction may have a high accuracy [5]. To simplify the problem, we assume that there is no prediction error in our validation methodology.

IV. VM SCHEDULING

A. Baseline algorithm

Here we describe the baseline algorithm which will be used as a comparison reference. Whether the renewable energy is sufficient for the workload energy or not, we expect to minimize the total energy consumption of the servers. The

minimization can be done at different levels such as infrastructure or application for instance. As mentioned previously, a server in idle state consumes roughly half of its peak power. Therefore, an effective approach consists in reducing the number of powered-on physical servers.

Given a set of VMs with different resources requirements and a set of servers with fixed capacities, we want to find the minimum number of servers needed to contain all VMs, such that the amount of VMs' resource requirements assigned to each server does not exceed its capacity. The VM placement problem can be modeled as a n-dimensional bin-packing with finite number of bins, where the different VM resource requirements can be modeled as different sizes of items and the various server's resources correspond to different bin sizes. In this paper, we consider CPU and RAM as the constraints for both servers and VMs. The VM placement problem then become a 2D bin-packing problem which is an NP-Hard problem. To solve this problem, we adopt the First Fit Decreasing (FFD) heuristic algorithm. A proof shows that for every list L , $FFD(L) \leq 11/9 OPT(L) + 1$ where $OPT(L)$ denotes the minimum number of bins that L can be all packed in and $FFD(L)$ denotes the number of bins used by FFD [23].

The regular FFD scheduling algorithm usually considers VM resource requirements as the resources' constraints for its placement. However, Cloud jobs typically have resource utilization levels well below their resource requirements on average over time [7]. The resource over-commitment technique increases the server utilization by considering lower bounds than the actual user requirements for allocating resources to VMs and thus, putting more VMs on a single server. As a consequence, when this lower bound is reached by each VM running on the server, some VMs have to be suspended or migrated in order to free resources. As resource over-commitment is widespread in Cloud infrastructures [16], [24], we combine FFD and resource over-commit in order to increase server resource usage and reduce the number of powered-on servers. However, as the over-commitment configuration can greatly increase both CPU and RAM utilization, it can lead to overload the server. Consequently, we will need to migrate the VMs from the over-loaded servers to others thus incurring an extra energy consumption and performance degradation. Hence, for these reasons, the jobs need to be provisioned for their peak draw by analyzing the history of jobs behaviors and seeking a safety over-commitment configuration. We study a real-world job trace and, in a similar way as our previous work [7], we set a reasonable over-commitment configuration avoiding RAM overloads.

We take into account of the VM creation and VM live migration energy overhead. Unlike the VM creation, the energy consumption of VM migration depends on the VM disk size and the number of dirty pages in RAM that impacts the migration time.

The considered baseline algorithm implements both FFD and over-commit resources techniques, as our opportunistic algorithm does. At any time, the jobs are submitted and

the broker directly places them on the servers. The baseline consumes the solar energy when it is available. The battery is charged when a surplus solar energy appears. Otherwise, the workload first discharges the battery and then uses the brown energy. Note that, there is no opportunistic job scheduling mechanism in the baseline algorithm.

B. Opportunistic job scheduling

As time in our system is discrete, the optimization operations are performed periodically at each slot. According to the job characteristics, opportunistic scheduling approaches classify the jobs into two types called here web jobs and batch jobs. The web job is defined as non-interruptible job. It has the highest priority on scheduling. When both types of jobs arrive, the broker pushes them respectively to the web job pool and then to the web queue, the batch job to the batch pool. We adopt the FFD algorithm to place the web jobs, all the jobs in web queue are immediately placed on the servers which have sufficient resources. Unlike the web jobs, batch jobs can be suspended with a slack time that may increase the potential chance to exploit the renewable energy. When the slack time reaches 0, the batch job is promoted as a web job. After all the web jobs have been placed, the broker seeks among the running servers which meet the batch jobs resource requirement. The web job placement and batch job placement are independent algorithms.

Recall that, opportunistic scheduling targets two problems: 1) when workload energy consumption is higher than the solar energy supply, it runs partially the workload: it suspends the batch jobs which have a non-null slack time and performs VM consolidation in order to switch-off more servers. This is to reduce the brown energy consumption. 2) Otherwise, it runs the entire workload and the batch jobs which were delayed before. This is to maximize the solar energy usage.

Due to the ESD efficiency, there is an energy consumption during battery charging. In contrary, opportunistic scheduling can reduce the effect due to battery efficiency by delaying the batch jobs to consume the solar energy directly instead of storing it in the battery. However, the more batch jobs are delayed, the more numerous VM migrations will be due to consolidation.

C. Battery charge/discharge model

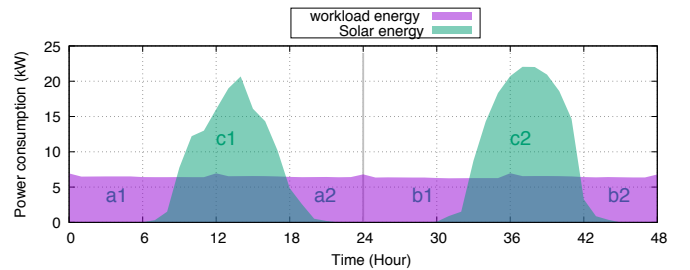


Fig. 2: Workload energy consumption and solar energy

As shown in Figure 2, the purple curve $w(t)$ denotes the workload energy consumption and the green curve $g(t)$

denotes the solar power. We observe in Figure. 2 that for areas a_1, a_2 and b_1, b_2 , the workload energy consumption is higher than the solar energy supply. If there is no ESD, the total energy consumption from the grid can be expressed as:

$$E_{\text{brown}} = \sum_{t=0}^{t_1} (w(t) - g(t)), \forall (w(t) > g(t)), t \in T \quad (2)$$

When the workload energy demand is less than the solar energy, the amount of surplus solar energy is defined as:

$$E_{\text{surplus}} = \sum_{t=t_1}^{t_2} (g(t) - w(t)), \forall (w(t) < g(t)), t \in T \quad (3)$$

For day 1 on Figure 2, $E_{\text{brown}} = a_1 + a_2$ and $E_{\text{surplus}} = c_1$. The battery has to be charged when $w(t) < g(t)$. When the solar energy is not sufficient to supply for the current workload energy, we first discharge the battery. Once the battery runs out, the servers then consume the brown energy from the grid.

In the rest of paper, we seek for the relationship between c_1 and $(a_2 + b_1)$ in different cases. Ideally, if c_1 is much larger than $(a_2 + b_1)$, the amount of energy produced by solar panels is sufficient to offset the whole workload energy consumption that takes into account an ideal ESD (it can store all the surplus solar energy). In a real case, the battery often has limited size and solar energy is not sufficient to compensate the workload energy consumption ($c_1 < (a_2 + b_1)$).

If we assume the ESD is ideal for both solutions, then in baseline, the energy loss is mainly caused by the battery efficiency. The solar energy needed can be formulated as:

$$c_1 \times \sigma > (a_2 + b_1) \quad (4)$$

where σ denotes the battery energy efficiency.

V. EXPERIMENTATION CONDITIONS

We developed a trace-driven simulator to compare different resource allocation and scheduling policies and estimate their energy consumption using a power model based on real measurements. The simulator integrates the data center and ESD models described in Section III. We use it to evaluate the impact on energy consumption with different configuration and ESD technologies. This section presents the experimentation conditions used in the simulator.

A. Workload trace

For all the simulations, we use the real-world trace from a medium-scale private Cloud data center provided by Easyvirt, a French SME which has a research partnership with this Cloud provider. The original trace was collected from 26th of March 2014 to 5th of July 2014. We extracted a non-holiday week: the data consists of 787 web jobs and 3148 batch jobs. It precises each job's initial VM resource requirement, the instantaneous CPU and RAM utilization. In our scenario, the CPU and RAM utilization is averaged over 1 hour in order to eliminate the noise. Each web job takes roughly 12 hours and each batch job takes about 6 hours with a deadline equal to 12 hours (it has to be executed within the 12 hours following its submission).

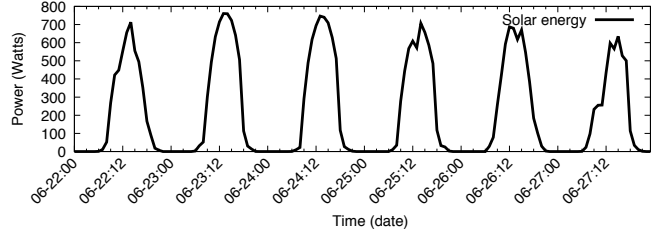


Fig. 3: Solar energy production with solar panels of 5.52 m²

B. Server energy consumption model

A server power consumption is related to its different components. Most previous studies [8] agree on the fact that the dynamic server power consumption mainly depends on the CPU frequency. We performed power consumption measurements on the Grid'5000 testbed [25]. The used servers are Dell PowerEdge R720 with two 6-cores Intel Xeon E5-2630 processors (2.3GHz), 32 GB of RAM and 600 GB of disk. The power consumption was taken for different CPU load profiles as described in [7]. Our experimental results shows in particular that a server on idle state can roughly consume half of its maximal power consumption.

C. Solar energy trace

For solar energy production, we use a mini-scale solar power farm which was set up in the campus of University Nantes¹. It is composed by 8 identical panels *Sanyo HIP-240-HDE4* and *SMA Sunny Boy 1200* inverter. The theoretical max power is 240 Watt per panel. Then the instantaneous peak power can be expressed as a function of m : e.g. for a given solar power farm with 8 panels, $\max E_{spv} = 240 \times m$ watt where $m = 8$. As shown in Figure 3, we choose the trace of a random week (22-28 June 2015) which is mostly sunny.

VI. RESULTS

In this section, we compare the opportunistic scheduling approach and the battery approach for maximizing solar energy utilization in a data center. First, we determine the optimal solar panel and the optimal battery size depending on the approach. Then we compare both approaches under various conditions, and we combine them.

A. Find the optimal solar panel dimension

We assume that the battery size and the charging/discharging rate both approach infinite in this early experiment to determine the optimal solar panel dimension for the given workload. We seek for a ideal solar panel that can supply the entire workload energy consumption. We first find a solution for this problem in the baseline case. Since the workload energy consumption can be estimated statistically, the area of solar panels can be trivially determined via calculation.

In this scenario, we assume an infinite battery size. So, it enables to store all the surplus solar energy in the region c_1 and c_2 as shown in Figure 2. Due to the battery efficiency

¹Traces available online: <http://photovolta2.univ-nantes.fr>

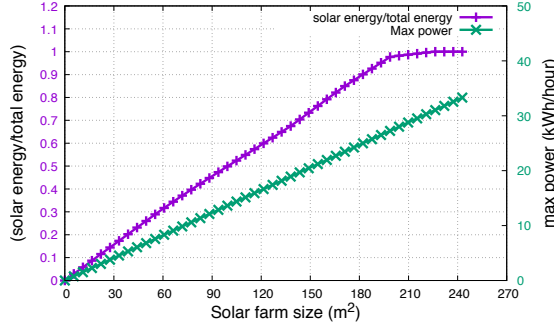


Fig. 4: Workload energy/solar energy supply ratio and maximum solar energy that can be generated per unit time

and limited charge/discharge rates, the amount of surplus solar energy has to be strictly greater than the required workload energy in order to compensate for these intrinsic ESD losses. Recall that, we are looking for c_1 which satisfies $c_1 \times \sigma \geq a_2 + b_1$ where σ indicates the battery efficiency. This formula describes the energy stored during the day which furnishes the workload energy until the solar energy becomes available again.

Figure 4 illustrates the workload energy/solar energy supply ratio. As the size of solar panels increase, the brown energy consumption of the entire workload decreases. When the solar panel area increases to 226.32 m² (about 15 × 15 meters), the workload energy consumption reaches 0, under the assumption of a battery with infinite size and infinite charging/discharging rate. We later determine the optimal battery size.

B. Find optimal battery size in ideal case

In this scenario, we assume that the size of solar panels are ideal, so equal to 226.32 m². Thus, it can provide sufficient energy for the workload needs. Figure 5 shows the brown energy consumption of baseline algorithm and of opportunistic scheduling with the same type of battery (LI battery) depending on the battery size (purple and green curves).

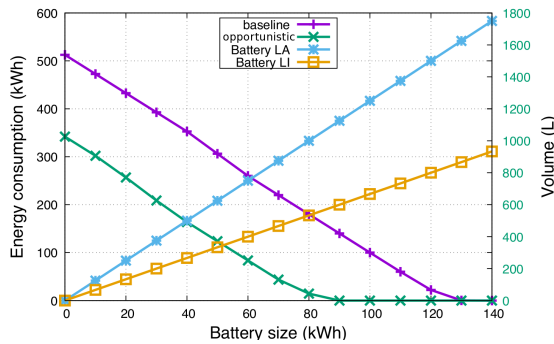


Fig. 5: First two curves (purple and green): brown energy consumption with an LI ESD. Last two curves (blue and yellow): ESD volume.

We also indicate the corresponding volume taken by the batteries depending on their capacity for both LI and LA

types (blue and yellow curves). The brown energy of both solutions decreases when we increase the battery size. We can see that opportunistic scheduling always requires a smaller battery size need than baseline, as expected. We can also see that the workload does not consume anymore brown energy when battery size is larger than 90 kWh (600 L for a LI ESD) for opportunistic and 140 kWh (950 L for a LI ESD) for baseline. So, using opportunistic scheduling decreases the ideal battery capacity and its corresponding volume by 36%.

With opportunistic scheduling, the batch jobs are delayed in order to be executed when solar energy is available. This has two effects: 1) a part of solar energy is directly consumed instead of storing it the battery; 2) despite the batch jobs delayed run, the surplus solar energy is reduced. This is why it can use a smaller battery size than baseline to reach the same brown energy consumption. Figure 5 also shows that LA volume is larger than LI. Using the price values provided in Table I, for a 90 kWh battery, for LI, it represents 600 L and 47,250 \$, while for LA, it represents 1,150 L and 18,000 \$. Both price and volume can impact the adoption of such an approach for logistics and financial reasons. While LI and LA batteries exhibit different characteristics, their loading and discharging schemes behave similarly. In the remaining of the paper, we use LI batteries without loss of generality.

C. Opportunistic vs. baseline when solar energy is not sufficient for the workload

In this scenario and for all the experiments presented below, we assume that there is not enough solar energy to fulfill the entire workload needs. We consider the solar panel dimension which is not able to provide sufficient solar energy to compensate the workload energy consumption. We compare the brown energy consumption with variable battery sizes for the both solutions, opportunistic and baseline.

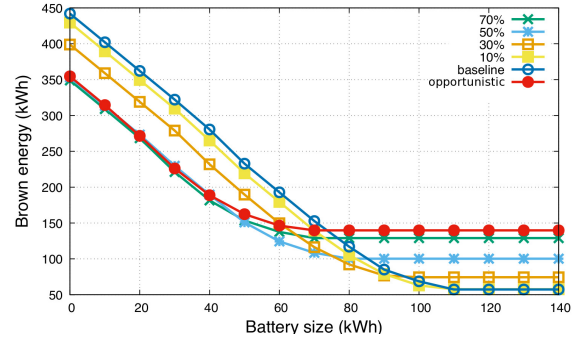


Fig. 6: Brown energy consumption with varying battery size

Figure 6 demonstrates the different battery sizes with multiple configuration for the opportunistic and baseline algorithms. The different configurations of the opportunistic scheduling are defined as delaying 30%, 50%, 70%, 90%, 100% batch jobs (instead of directly executing them when solar energy is available, and thus storing it). This represents possible trade-off between the opportunistic scheduling approach and the ESD-based approach. We can see that the both opportunistic

and baseline reduce brown energy when the battery size becomes large. For a given battery size which is inferior to 73 kWh, opportunistic always consumes less brown energy in comparison with baseline. After this point, the brown energy consumption of opportunistic approach does not decrease any more when battery size becomes larger. In contrast, the baseline brown consumption continues to decrease while the battery size increases up to 110 kWh. Although all the other configurations of opportunistic approach can get lower brown energy with a larger battery compared with pure opportunistic scheduling, they are still higher than baseline in this particular case.

D. Solar energy losses with variable battery size

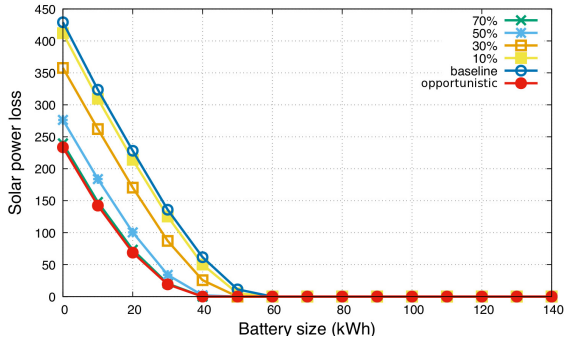


Fig. 7: Solar energy lost due to the limited battery size

Figure 7 shows the energy losses with varying battery sizes. This energy losses stem from the battery limited charging rate and size. Since the opportunistic approach delays the batch jobs until the solar energy becomes available, the workload here consumes solar energy directly and stores the rest of solar energy to the battery. Thus, opportunistic can use a smaller battery while achieving the equivalent effect as baseline. Finally, when battery size is 80 kWh for opportunistic and 110 kWh for baseline, they both reach zero solar energy loss caused by the battery’s limited charging rate and size.

E. Opportunistic scheduling migration vs. baseline battery loss

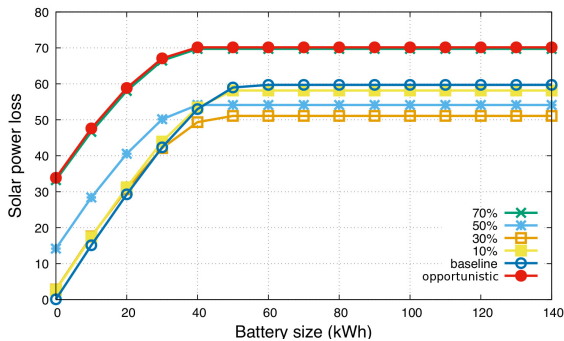


Fig. 8: Migration cost vs. battery efficiency loss

Figure 8 shows the amount of energy lost with variable battery size. Here, we focus on 2 types of energy loss: 1) battery energy efficiency; 2) VM migration energy cost. In baseline, as the size of battery increases, the energy loss increases; this happens because of battery energy efficiency and of very few VM migrations due to overloaded servers (over-commit policy). Hence, the choice of battery type highly impact the waste of energy.

For the opportunistic approach, the energy loss mainly depends on: 1) migrations caused by consolidation; 2) the rest is same as baseline that is the battery efficiency. Since the solar energy was not sufficient enough for the workload needs, the opportunistic algorithm has to suspend some batch jobs and to perform consolidation in order to keep a low number of powered-on servers. And the delayed batch jobs then are executed when solar energy become available again. The delayed workload directly consumes the solar energy and the remaining solar energy is stored in the battery. Thus the opportunistic approach stores less energy than baseline in the ESD, and thus the loss due to battery efficiency are lower with the opportunistic approach. However, the total solar energy is not sufficient for the entire workload in this case, the opportunistic approach periodically performs VM consolidations that may lead to a number of VM migrations. This migration energy cost compensates that gain. For this reason, it can be better to partially delay the batch jobs (respectively 10, 30, 50 and 70% are delayed). In fact, when we delay less batch jobs, it leads to less migrations by consolidation, but more energy will be stored in the ESD. There is a balance for the opportunistic approach between the energy loss caused by migrations and by battery efficiency. We observe that when there are 30% batch jobs delayed, the energy loss which contains the migration energy overhead and battery efficiency, the opportunistic approach gets a lower energy loss than baseline when battery size is greater than 40 kWh.

F. FFD scheduling impact

For a given workload such as ours, the opportunistic approach has a lower energy loss in comparison with baseline when it delays 30% batch jobs. Unfortunately, in Figure 6, the opportunistic approach consumes more brown energy than baseline. It seems that the result is in conflict with the result in Figure 8. After our analysis, there is also an impact from the two consecutive FFD algorithms (for web and batch jobs) that leads the opportunistic approach to need more servers to place the same amount of jobs compared with baseline.

There is a performance degradation when we change the input list size; e.g. for any list L with a length l , $FFD(L) \leq 11/9OPT(L) + 1$ as mentioned in Section IV-A. If we divide the list L into 2 sub-lists L_1 with a length $0 < l_1 \leq n - t$ and L_2 with length $0 < l_2 \leq t$, $FFD(L_1) \leq 11/9OPT(L_1) + 1$ and $FFD(L_2) \leq OPT(L_2) + 1$. The performance may be different between $FFD(L)$ and $(FFD(L_1) + FFD(L_2))$. In case of our simulation results, we observe that there is a performance degradation with the number of bins needed:

$FFD(L) \ll (FFD(L_1) + FFD(L_2))$. This explains why the opportunistic scheduling has a lower energy loss than baseline in Figure 8, but it consumes more brown energy than baseline in Figure 6 when the battery size is large enough.

VII. CONCLUSION

Integrating renewable energy into data centers significantly reduces the traditional energy consumption and carbon footprint of these energy-hungry infrastructures. As renewable energy is intermittent and fluctuates with time, it is usually under-utilized. In this paper, we address the problem of improving the utilization of renewable energy for a single data center by using 2 approaches: opportunistic scheduling and energy storage. Our first result deals with analyzing the workload to find ideal solar panel dimension and battery size, this is used to power the entire workload without any brown energy consumption. However, in reality, either the solar panel dimension or the battery size are limited, and we still have to address the problem of matching the workload consumption and renewable energy production. The second result shows that opportunistic scheduling can reduce the demand for battery size while the renewable energy is sufficient. The last results demonstrate that for different battery sizes and solar panel dimensions, we can find an optimal solution combining both approaches that balances the energy losses due to different causes such as battery efficiency and VM migrations due to consolidation algorithms. Our future work includes studying the pertinence of both approaches with other renewable energy sources, like wind for instance. As wind energy presents a completely different production profile compared to solar energy, we would like to investigate whether the trade-off between the opportunistic scheduling and energy storage approaches which is proposed in this paper remains the same or would be different.

ACKNOWLEDGMENT

This work has received a French state support granted to the CominLabs excellence laboratory and managed by the National Research Agency in the "Investing for the Future" program under reference Nb. ANR-10-LABX-07-01.

REFERENCES

- [1] J. Glanz, "Power, Pollution and the Internet," *The New York Times*, 2012.
- [2] L. Research, "Coal Computing: How Companies Misunderstand Their Dirty Data Centers," White paper, 2016.
- [3] "How dirty is your data?" Greenpeace report, 2011.
- [4] J. Whitney and P. Delforge, "Data Center Efficiency Assessment scaling up energy efficiency across the Data Center Industry: evaluating Key Drivers and Barriers," NRDC white paper, 2014.
- [5] I. n. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, "Parasol and GreenSwitch: Managing Datacenters Powered by Renewable Energy," in *ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2013, pp. 51–64.
- [6] S. Figuerola, M. Lemay, V. Reijs, M. Savoie, and B. St. Arnaud, "Converged Optical Network Infrastructures in Support of Future Internet and Grid Services Using IaaS to Reduce GHG Emissions," *Journal of Lightwave Technology*, vol. 27, no. 12, pp. 1941–1946, 2009.
- [7] Y. Li, A.-C. Orgerie, and J.-M. Menaud, "Opportunistic Scheduling in Clouds Partially Powered by Green Energy," in *IEEE International Conference on Green Computing and Communications (GreenCom)*, Sydney, Australia, Dec. 2015.
- [8] A.-C. Orgerie, M. Dias de Assunção, and L. Lefèvre, "A survey on techniques for improving the energy efficiency of large-scale distributed systems," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 47, 2014.
- [9] R. Buyya, A. Beloglazov, and J. Abawajy, "Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges," *Int. Conf. on Parallel and Distributed Processing Techniques and Applications (PDPTA)*, 2010.
- [10] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," *Future generation computer systems*, vol. 28, no. 5, pp. 755–768, 2012.
- [11] S. Srikantaiah, A. Kansal, and F. Zhao, "Energy aware consolidation for cloud computing," in *Conference on Power aware computing and systems*, 2008, pp. 1–5.
- [12] C. Ghribi, M. Hadji, and D. Zeghlache, "Energy efficient vm scheduling for cloud data centers: Exact allocation and migration algorithms," in *IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 2013, pp. 671–678.
- [13] L. Wang, G. Von Laszewski, J. Dayal, and F. Wang, "Towards energy aware scheduling for precedence constrained parallel tasks in a cluster with dvfs," in *Cluster, Cloud and Grid Computing (CCGrid), 2010 10th IEEE/ACM International Conference on*. IEEE, 2010, pp. 368–377.
- [14] A. Gandhi, M. Harchol-Balter, R. Das, and C. Lefurgy, "Optimal power allocation in server farms," in *ACM SIGMETRICS Perf. Evaluation Review*, vol. 37, no. 1, 2009, pp. 157–168.
- [15] M. Etinski, J. Corbalán, J. Labarta, and M. Valero, "Understanding the future of energy-performance trade-off via dvfs in hpc environments," *Journal of Parallel and Distributed Computing*, vol. 72, no. 4, 2012.
- [16] X. Zhang, Z.-Y. Shae, S. Zheng, and H. Jamjoom, "Virtual machine migration in an over-committed cloud," in *IEEE Network Operations and Management Symposium (NOMS)*, 2012, pp. 196–203.
- [17] A. Krioukov, C. Goebel, S. Alspaugh, Y. Chen, D. E. Culler, and R. H. Katz, "Integrating renewable energy using data analytics systems: Challenges and opportunities." *IEEE Data Eng. Bull.*, vol. 34, no. 1, pp. 3–11, 2011.
- [18] Y. Ghiassi-Farrokhfal, S. Keshav, and C. Rosenberg, "Toward a realistic performance analysis of storage systems in smart grids," *IEEE Transactions on Smart Grid*, vol. 6, no. 1, pp. 402–410, 2015.
- [19] H. Chen, T. N. Cong, W. Yang, C. Tan, Y. Li, and Y. Ding, "Progress in electrical energy storage system: A critical review," *Progress in Natural Science*, vol. 19, no. 3, pp. 291–312, 2009.
- [20] K. Divya and J. Østergaard, "Battery energy storage technology for power systems?an overview," *Electric Power Systems Research*, vol. 79, no. 4, pp. 511–520, 2009.
- [21] D. Wang, C. Ren, A. Sivasubramaniam, B. Urgaonkar, and H. Fathy, "Energy storage in datacenters: what, where, and how much?," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 40, no. 1, 2012, pp. 187–198.
- [22] N. Beldiceanu, B. Dumas Feris, P. Gravey, S. Hasan, C. Jard, T. Ledoux, Y. Li, D. Lime, G. Madi-Wamba, J.-M. Menaud, P. Morel, M. Morvan, M.-L. Moulinard, A.-C. Orgerie, J.-L. Papat, O. H. Roux, and A. Sharaiha, "Towards energy-proportional Clouds partially powered by renewable energy," *Computing*, p. 20, 2016.
- [23] M. Yue, "A simple proof of the inequality $ffd(l)? 11/9 opt(l)+ 1,? 1$ for the ffd bin-packing algorithm," *Acta mathematica applicatae sinica*, vol. 7, no. 4, pp. 321–331, 1991.
- [24] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, "Efficient datacenter resource utilization through cloud resource overcommitment," in *IEEE Conf. on Computer Communications Workshops (INFOCOM Workshops)*, 2015, pp. 330–335.
- [25] D. Balouek, A. Carpen Amarie, G. Charrier, F. Desprez, E. Jeannot, E. Jeanvoine, A. Lèbre, D. Margery, N. Niclausse, L. Nussbaum, O. Richard, C. Pérez, F. Quesnel, C. Rohr, and L. Sarzyniec, "Adding virtualization capabilities to the Grid'5000 testbed," in *Cloud Computing and Services Science*. Springer International Publishing, 2013, vol. 367, pp. 3–20.