Received 30 September 2017; | Revised 31 December 2017; | Accepted 14 February 2018

DOI: xxx/xxxx

CONCURRENCY & COMPUTATION: PRACTICE & EXPERIENCE

Quantifying the Impact of Shutdown Techniques for Energy-Efficient Data Centers

Issam Rais*[1,2] | Anne-Cécile Orgerie[2] | Martin Quinson[2] | Laurent Lefèvre[1]

[1]Avalon Team, LIP (Inria, ENS Lyon, CNRS, Univ. Lyon 1), Lyon, France
[2]Univ. Rennes, Inria, CNRS, IRISA, Rennes, France

Correspondence
*Issam Rais. Email: issam.rais@inria.fr

Present Address
ENS de Lyon
46 allée d'Italie
69364 Lyon - France

Summary

Current large-scale systems, like datacenters and supercomputers, are facing an increasing electricity consumption. These infrastructures are often dimensioned according to the workload peak. However, as their consumption is not power-proportional: when the workload is low, the power consumption is still high. Shutdown techniques have been developed to adapt the number of switched-on servers to the actual workload. However, datacenter operators are reluctant to adopt such approaches because of their potential impact on reactivity and hardware failures, and their energy gain which is often largely misjudged. In this article, we evaluate the potential gain of shutdown techniques by taking into account shutdown and boot up costs in time and energy. This evaluation is made on recent server architectures and future energy-aware architectures. Our simulations exploit real traces collected on production infrastructures under various machine configurations with several shutdown policies, with and without workload prediction. We study the impact of future's knowledge for saving energy with such policies. Finally, we examine the energy benefits brought by suspend-to-disk and suspend-to-RAM techniques and we study the impact of shutdown techniques on the energy consumption of prospective hardware with heterogeneous processors (big-medium-little paradigm).

KEYWORDS:
Energy efficiency, data centers, shutdown techniques, energy-aware hardware, sleep modes

1 | INTRODUCTION

Data centers are responsible for about 2% of global carbon emissions today and use 80 million megawatt-hours of energy annually, almost 1.5 times the amount of electricity used by the whole of New York City (1). This tremendous energy consumption is becoming more and more concerning and will even worsen with the rapid growth of Cloud computing infrastructures and the raise of a new generation of smart devices requiring always more data centers to provide new services.

In order to make data centers more energy-efficient, a wide variety of approaches have been proposed in the recent years, ranging from free cooling to low-power processors, and tackling wasted watts at each level of the data center (2). One of the most promising solution for data centers consists in consolidating the workload on an optimized number of servers and switching off idle servers (3, 4, 5, 6). While such an on/off approach has been extensively studied in literature, most infrastructure administrators still dare not use it in their datacenters. This situation is due to two factors: firstly, until very recently, servers were not designed to be switched off; secondly, switching off takes time and energy. So it is difficult for administrators to estimate their potential energy gains versus their potential loss of reactivity due to a too long booting time. Several solutions have been proposed to limit this possible performance impact, like keeping few nodes idle (reactivity margin) or using hibernation or standby modes to fasten the boot.

In this paper, we study different shutdown techniques for computing resources in data centers, like actual switching off and hibernation modes. Moreover, we estimate the impact of such techniques on the energy consumption, the reactivity of the platform and on the lifetime of the servers. Our validations combine real power measurements and real datacenter traces with simulation tools in order to also study future energy-efficient modes and future energy-aware server architectures.

In particular, this paper extends our previous work presented in (7) with studies on 1) future energy-proportional architectures using the big-medium-little paradigm, 2) reactivity margin for less aggressive shutdown policies, and 3) alternative techniques to complete shutdown, like suspend-to-disk and suspend-to-ram.

Our simulations, based on real power measurements and large-scale workload traces from production and experimental platforms, show that shutdown techniques can save at least 84% of the energy that would be otherwise lost to power idle nodes. This conclusion remains true for prospective energy-proportional hardware and even aggressive shutdown policies do not have impact on hardware lifetime. Our results also show that from an energy perspective, efforts should focus on reducing the energy consumption while in sleep mode instead of trying to reduce the time it takes to switch between on and off modes.

This paper makes the following contributions:

1. an analysis of the impacts of shutdown policies (ie. switching off unused servers) on disk lifetime and on the energy consumption of production infrastructures over long periods through simulation of real platform traces: 15 months of a scientific cluster and 6 years of an experimental testbed.
2. a study of the energy savings reachable through classical shutdown policies, and the influence of workload prediction and reactivity margin (keeping part of the servers on although they are idle for reactivity reasons) on these savings;
3. an estimation of the energy gains of shutdown policies on future hardware, including power-proportional architectures and servers with fast and reliable standby mode (like suspend-to-disk and suspend-to-RAM);

The reminder of this paper is structured as follows. Section 2 presents the related work. The on/off energy model and the shutdown policies are introduced in Section 3. The experimental setup is provided in Section 4. The experimental validation is shown in Section 5 for current hardware and in Section 6 for future architectures. Finally, Section 7 concludes this work and presents the future directions.

2 | RELATED WORK

Shutdown techniques require 1) the hardware ability to remotely switch on and off servers, and 2) energy-aware algorithms to timely employ such an ability. This section describes the state-of-the-art approaches for both features.

2.1 | ACPI specification

ACPI (Advanced Configuration and Power Interface) is the key point for implementing Operating System-directed configuration and Power Management (OSPM). Hardware and software vendors are then encouraged to implement and respect these defined interfaces. ACPI specifies different power states for several kinds of devices, among which the C-states concerning processor power consumption and thermal management for instance.

In this study on shutdown approaches, we focus on the ACPI sleeping states. They consists in various types of node configurations including different sleeping policies and protocols. According to the ACPI specification, there are 5 possibles sleeping state:

- S1 (weak): this is a low wake latency sleeping. No system context is lost and no hardware is turned off in this state.
- S2 (weak): Similar to S1, except that CPU and system cache is lost
- S3: low wake latency sleeping state where all context is lost except system memory i.e CPU, caches, and chip set context are lost. Hardware restores some CPU and L2 configurations and maintains memory context.
- S4: Lowest power but longest wake latency with only platform context is maintained.
- S5: System shutdown state. Similar to S4, except that the OS doesn't save any context.

2.2 | Suspend modes on Linux kernel

Given this theoretical specification, we focus on the Linux implementation of this system power management. The available sleep states on the Linux kernel are:

- S0 or "Suspend to Idle" : freezing user space and putting all I/O devices into low-power states
- S1 or "Standby / Power-On Suspend" : same as S0 adding the fact that non boot CPUs are put in offline mode and all low-level systems functions are suspended during transitions into this state. The CPU retains power meaning operating state is lost, so the system easily starts up again where it left off
- S3 or "Suspend-to-RAM" : Everything in the system is put into low power state mode. System and device state is saved and kept in memory (RAM).
- S4 or "Suspend-to-disk" : Like S3, adding a final step of writing memory contents to disk.

On the top of our knowledge, many datacenters servers do not implement or allow S3 (Suspend-to-RAM) sleep state, because of numerous errors when resuming (especially errors due to network connections with Myrinet or Ethernet protocols). Typically, only S0, S4 and S5 are available for operational use.

2.3 | Shutdown policies

Early work studying the energy-related impacts of shutdown techniques started in 2001 (8, 9). Yet, they do not consider any transition cost for switching between on and off, but they nonetheless showed the potential impact of such techniques (10). Demaine *et al.* examine the power minimization problem where the objective is to minimize the total transition costs plus the total time spent in the active state (11). They develop a $(1 + 2\alpha)$-approximation algorithm, with α the transition cost.

However, the parameters considered for this transition cost highly varies across the literature (10). Gandhi *et al.* take into account the energy cost of switching on servers (no switching off cost as it is estimated to be negligible in comparison with the switching on cost) (12). This energy cost is assumed to be equal to the transition time multiplied by the power consumption while in the on state. Lin *et al.* take into account the energy used for the transition, the delay in migrating connections or data, the increased wear-and-tear on the servers, and the risk associated with server toggling (13).

Shutdown policies are nowadays easily implementable with off-the-shelf hardware. In fact, most of data center resource managers propose techniques or hooks to configure such capabilities. For example, slurm (14), an open-source cluster management system introduces a *SuspendTime*[1] that represents the minimum idle time after which it allows the node to be switched off.

Then, the resource manager is responsible for deciding when to suspend and resume nodes. It takes decisions either based on pre-determined policy (14) or on workload predictions (3). Shutdown policies are often combined with consolidation algorithms which gather the load on less nodes to favor the shutdown of more nodes. Employing either reactive or proactive scheduling options (6, 15), consolidation algorithms increase the energy gains brought by shutdown techniques at a cost of a trade-off with performance (5). In this paper, we study simple shutdown techniques, without combining them to scheduling algorithms and consolidation approaches in order to evaluate the impacts of such techniques without interfering with the workload of real platforms and with the users' expected performances.

The main disadvantage of shutdown policies resides in the energy and time losses that may occur when switching off and on takes longer than the actual idle period. In such cases, the user has to wait for the server to be on (up to few minutes), thus incurring a lessening in the platform rapidity in handling incoming users requests (16). The various suspend modes offer different performances concerning the time they need to switch between the On and Off states and the energy they consume while in Off state. The next section provides the necessary formalism for evaluating the impact of such key parameters for shutdown techniques.

Shutdown policies' impacts also extend to temperature management in data centers, and consequently cooling systems, as they abruptly clear or add heat sources (i.e. servers) (17). At a data center level, it translates into power budgeting, where the total power budget is partitioned among the cooling and computing units, and where the cooling power has to be sufficient to extract the heat of the computing power (10). Given the computing power budget, Zhan *et al.* propose an optimal computing budgeting technique based on knapsack-solving algorithms to determine the power caps for the individual servers (18). Adding constraints (i.e. temperature, power capping or energy budget) lead to the design of complex shutdown policies where energy consumption belongs to the multiple criteria to optimize, but is generally not the predominant one (10).

[1] http://slurm.schedmd.com/power_save.html

3 | MODELS

In this section, we describe the different models used by the shutdown policies we want to evaluate in order to determine when a node has to be switched off.

3.1 | Energy efficiency time threshold

Switching on and off a server consumes time and energy, it is thus required to take these costs into account when deciding whether to switch off an idle server or not.

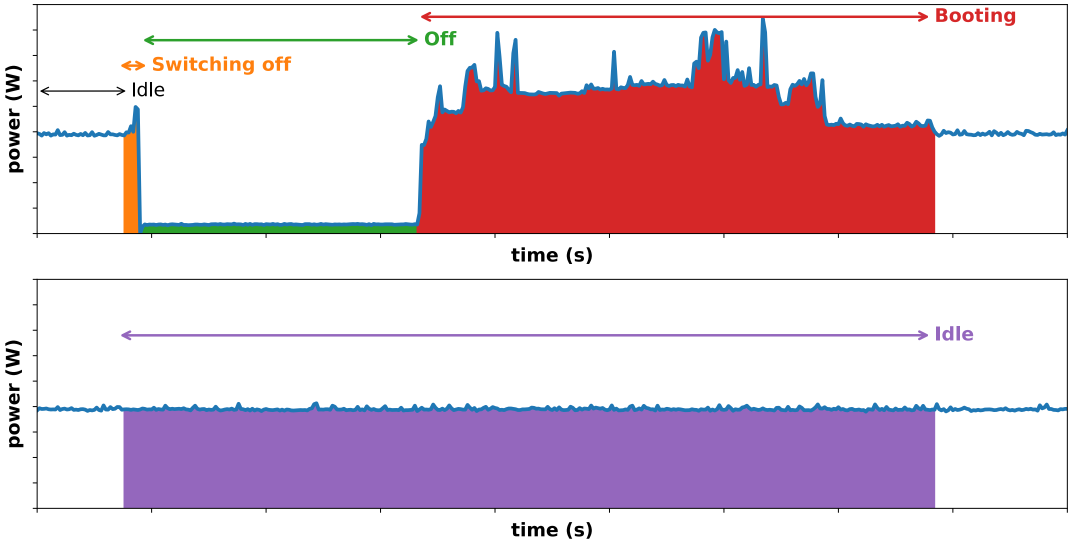


FIGURE 1 Time threshold to decide whether to switch off or not

In (16), the authors introduce T_s a time threshold such that: when a node is idle for more than T_s seconds, it is more energy-efficient to switch it off and then on again at the adequate time; otherwise, if the server is idle for less than T_s, it should remain idle to save energy. Moreover, T_s needs to be greater than the time required to switch off and on again a server in order for this threshold to be physically acceptable. Figure 1 illustrates the computation of this T_s time threshold. On both graphs, the blue curve depicts the power consumption of a machine over time. The colored areas of these two graphs correspond to the energy consumed in the two cases. The upper graph represents a machine where an On to Off sequence is launched (the orange area highlights the energy consumption during this sequence), followed by an Off section (green area), and then an Off to On sequence (red area). The bottom graph represents the same machine in Idle state for the same time period (purple area). So, T_s is the time threshold such that the areas of both graphs (orange + green + red in the first case, and purple in the second case) are equal.

Following this principle, T_s is defined as:

$$T_s = \mathbf{max}\left(\frac{E_{OnOff} + E_{OffOn} - P_{Off}(T_{OnOff} + T_{OffOn})}{P_{idle} - P_{Off}}, (T_{OnOff} + T_{OffOn})\right)$$

where:

- P_{idle} is the power consumption when the node is unused, but powered on;
- P_{off} is the power consumption when the node is switched off (typically not null and lower than P_{idle});
- T_{OnOff} is the time spent by the node when asked for a On-Off sequence;
- T_{OffOn} is the time spent by the node when asked for a Off-On sequence;
- E_{OnOff} is the energy consumed during the On-Off sequence;

- E_{OffOn} is the energy consumed during the Off-On sequence.

In order to compute T_s, all parameters have to be known for each concerned server. These parameters can be acquired through a calibration measurement campaign. Then a shutdown policy is required to know when to switch off nodes. Indeed, as future is not known in the general case, it is difficult to determine for a given idle data center server if it will stay idle for more than T_s or not.

As explained in Section 2.3, energy consumption can be combined with other constraints (i.e. power capping or temperature) to design more complex shutdown policies. Yet, the time threshold model used in this paper is the simplest one guaranteeing energy savings (by definition of T_s), and therefore, it provides a solid basis for more elaborated models.

3.2 | Studied shutdown policies

As the goal of this paper is to evaluate the impacts of on/off strategies rather than proposing new shutdown policies, we chose to lean on two ideal policies which will provide theoretical values about energy consumption.

Policy P1: knowing the future In this first policy, we consider that the future is completely known. Thus, dates and lengths of idle period are known for each server.

This policy will give a theoretical lower bound for energy consumption with a perfect prediction algorithm. Trace-driven experiments shown in (16) indicate a 7% difference on the overall energy consumption with a simple prediction algorithms compared to this perfect case (future known), so not far from the bound.

Policy P2: aggressive shutdown The second policy does not consider the future and tries to switch off a server as soon as it is in idle state without any prediction attempt. Such an aggressive approach is expected to result in a higher energy consumption than Policy 1 because some idle periods may be lower than T_s. In such cases, switching off increases the energy consumption compared to staying idle.

This policy provides a simplified version of actual algorithms that wait for a given amount of time (usually greater than T_s) before switching off idle nodes. For instance, a similar policy is implemented on the French experimental testbed Grid'5000 (19) with an additional measure: a small portion of idle nodes are kept idle to support urgent computation needs.

These two policies depict a representative sample of typical shutdown policies deployed on real data centers. They will be compared in order to provide an evaluation of the potential impacts of such policies on energy consumption and nodes lifetime.

4 | EXPERIMENT SETUP

In order to provide a fair comparison among policies P1 and P2, we simulate their behavior on real workload traces. The simulation tool is using calibration measurements that we performed on several servers representing different hardware architectures. Simulations combine the workload traces and the energy calibration values to compare the two policies according to relevant metrics presented at the end of this section.

4.1 | Workload traces

The utilized workload traces come from two kinds of data centers: an experimental small-size data center of an experimental testbed and a supercomputer for bioinformatics. They provide two different utilization scenarios which exhibit different workload patterns and utilization levels.

4.1.1 | Operational Cloud platform: E-Biothon

The E-Biothon platform is an experimental Cloud platform to help speed up and advance research in biology, health and environment (20). It is based on four Blue Gene/P racks and a web portal that allow members of the bioinformatics community to easily launch their scientific applications. Overall, the platform offers 4096 4-cores nodes, reaching a peak power of 56 Travel (20).

We obtained a workload trace for this platform covering from the 1st of January 2015 to the 1st of April 2016, so roughly 15 months of resource utilization. In this trace, the average size of idle periods is around 2.8 hours while the overall usage is 98%

4.1.2 | Experimental testbed: Grid'5000

Grid'5000 is a large-scale and versatile testbed for experiment-driven research in all areas of computer science, with a focus on parallel and distributed computing including Cloud, HPC and Big Data (19). Since 2005, the testbed offers distributed computing resources which are highly reconfigurable. Thus, it is a unique operational platform dedicated to experiments. In 2016, it consists of about 1,000 servers embedding 8,000 overall, geographically distributed on 9 sites. For our evaluation, we extracted the workload trace of the Rennes site from the 1st of April 2010 to the 1st of April 2016, thus representing 6 years of resource utilization on this site. During this period, the weighted arithmetic mean of the number of nodes is 149 and the average size of idle periods is around 6.17 hours. The overall usage is around 33%.

4.2 | Energy calibration for real servers

Along with computing nodes, Grid'5000 provides management tools like kapower3, a utility that allows a user to have control on the power status of a reserved node[2], and, on some sites, it gives access to external wattmeters monitoring entire servers and providing one power measurement per second per server with a 0.125 Watts accuracy. This infrastructure is used for obtaining the energy calibration measurements required to compute T_s as described in Section 3.1.

We applied the following protocol for calibration: an idle period of 20 seconds, followed by a monitored On to Off sequence, then a 20 seconds Off section, and finally a monitored Off to On sequence. Thus, every energy and time monitored sequence is followed and preceded by idle periods to avoid noise.

Grid'5000 servers are representative of typical architectures that can be found in usual datacenters. As it is an experimental testbed mainly for distributed systems research, it presents a high variety of servers (currently 24 different clusters). We used servers from three clusters which are power-monitored (one server per cluster). These servers present heterogeneous characteristics described on top of the Table 1 .

TABLE 1 Calibration nodes' characteristics and energy parameters for On-Off and Off-On sequences (average and standard deviation on 100 experimental measurements)

Features	Orion	Taurus	Paravance
Server model	Dell PowerEdge R720	Dell PowerEdge R720	Dell PowerEdge R630
CPU model	Intel Xeon E5-2630	Intel Xeon E5-2630	Intel Xeon E5-2630v3
# of CPU	2	2	2
Cores per CPU	6	6	8
Memory (GB)	32	32	128
Storage (GB)	2 x 300 (HDD)	2 x 300 (HDD)	2 x 600 (HDD)
GPU	Nvidia Tesla M2075	-	-

Parameters	Orion		Taurus		Paravance	
	Average	Std dev.	Average	Std dev.	Average	Std dev.
E_{OffOn} (J)	23,386	215.45	19,000	169.6	19,893	1,571.2
E_{OnOff} (J)	775.79	125.6	616.08	75.23	1,115	82.3
T_{OffOn} (s)	150	1.73	150	1.49	167.5	16.6
T_{OnOff} (s)	6.1	1.0	6.1	0.7	13	1.9
P_{idle} (W)	135	0.5	95	0.4	150	0.9
P_{off} (W)	18.5	0.4	8.5	0.3	4.5	0.6

T_s (s)	182.60	211.43	138.80

To obtain the needed calibration values, we monitored three servers (Orion, Taurus, Paravance) while performing switching off and on operations. The nodes are running a standard Debian Jessie (Debian GNU/Linux 8.0 for x64 architectures). The results presented on the bottom part of Table 1 show averaged values over 100 experiments for these operations with the S5 mode (regular shutdown). They provide all the requested values to compute T_s for the three nodes. Note that the observed standard deviations are negligible, thus confirming the stability of our measurements.

We performed similar experiments for the S4 mode (Suspend-to-Disk). However, our experiments show that S4 mode takes more time to switch between On and Off states (T_{OnOff}) than the S5, the same time for switching between Off and On (T_{OffOn}), and a similar Off power consumption (P_{Off}). Consequently, this mode is useless from an energy point of view. The nodes do not support the S3 mode (Suspend-to-RAM).

[2] https://www.grid5000.fr/mediawiki/index.php/Power_State_Manipulation_commands

4.3 | Energy calibration for future architectures

The ARM big.LITTLE processor is an example of promising architecture from an energy point of view. It combines a low-power processor with a high-performance one to offer an heterogeneous architecture closer to power proportionality than other processors even with dynamic frequency scaling (21). The idea consists in activating one processor at a time: either the low-power one during low workload or the powerful one during high activity.

This concept has been extended to data centers in (22) in order to build power-proportional servers. In this approach named BML (Big, Medium, Little), a computing node is composed of three processing units aiming at different levels of workload and energy consumption. It is assumed that each processing unit is able to be turned on and off independently from the others.

We take the same BML configuration that the calibration measurements presented in (23). A Big unit corresponds to Graphene node (x86 Intel Xeon X3440) on Grid'5000 platform, a Medium unit corresponds to a Chromebook (ARM Cortex-A15) and finally, a Little unit corresponds to a Raspberry node (ARM Cortex-A7). We then assume, as in (23), that it exists a computing node composed with these three processing units. The required energy values for BML nodes are provided in Table 2 . Interestingly, the Medium unit presents a behavior different from the two others with a $T_{OffOn} < T_{OnOff}$.

TABLE 2 Initial calibration values for independent BML units from (23)

Parameters	Big	Medium	Little
E_{OffOn} (Joules)	4,940	49.3	40.5
E_{OnOff} (Joules)	760	77.6	36.2
T_{OffOn} (seconds)	71	12	16
T_{OnOff} (seconds)	16	21	14
P_{idle} (Watts)	47.7	4	3.1
P_{off} (Watts)	8	1.9	2.2

We consider several configurations where BML components can be turned off separately or simultaneously. The considered configurations are the following:

- *AtomicBML*: the three processing units composing a BML node are turned off simultaneously, behaving as a single node. For this configuration, the energy values E_{OffOn} and E_{OnOff} are the sum of the three units' values, the times T_{OffOn} and T_{OnOff} are the maximum as we assume that components can be switched in parallel, and the power values P_{idle} and P_{off} are the sum.
- *OnlyB*, *OnlyM*, *OnlyL*: only one of the processing unit composing the BML node is turned off. In this case, P_{off} corresponds to the sum of the P_{off} of the turned off processing unit and the P_{idle} of the others. P_{idle} corresponds to the sum of the three P_{idle}. T_{OffOn}, T_{OnOff}, E_{OffOn} and E_{OnOff} are equal to the values in Table 2 of the component which is switched on or off.
- *FlexibleBML*: all possible computing units are turned off individually: if an idle period does not allow for all processing units to be turned off, only the possible ones are turned off.

These use cases represent possible configurations of future processing architectures which should get closer to energy proportionality than current ones. One can wonder if shutdown techniques can be beneficial for such architectures from an energy point of view.

4.4 | Evaluation metrics

In order to fairly compare the shutdown policies in the determined use cases, we define several evaluation metrics. In particular, for evaluating their energy impact, we compare the energy consumed with each policy against the energy used without any shutdown policy (ie. policy where the nodes stays idle and consumes P_{idle} Watts during periods without any work). This metric will indicate the potential energy savings with each policy.

We also provide the theoretical maximum energy savings if switching operations had a null cost (ie. zero energy, zero time for switching between on and off states). This provides an idea on how far the policies are from the theoretical ideal case and how much the costs related to switching operations are impacting the energy savings. The ideal case does not provide 100% energy gains compared to the idle case as switched off nodes consume energy ($P_{off} \neq 0$).

Finally, the results include the number of On-Off cycles per node for each workload in order to evaluate the impact of shutdown policies on the servers' lifetime. Indeed, one obstacle to the adoption of shutdown policies lies in the number of On-Off cycles imposed to the servers. In case of a too high number of cycles, it could damage the hardware parts like the hard disk drives (HDD). Typically, it is considered that hard drives can support a given amount of switching on and off during their lifetime. This parameter, known as *Contact Start/Stop Cycles* or *load/unload cycles* depending on the physical configuration of the hard drive head, is typically around 50,000 and 300,000 respectively for desktop HDD (24), and around 600,000 for NAS HDD (Network-Attached Storage) which use only load/unload technology (25). So, the number of On-Off cycles per node will be compared with these figures to determine whether the policy may alter or not the servers' lifetime.

Here we only consider HDD disks as they represent the dominant storage technology in current data centers (26). While solid state drives (SSD) start to equip data centers and provide higher data transfer rate and lower access latency than HDD, their high price and limited numbers of erasure cycles prevent them from being widely applied to large data centers (27). Yet, SSD technologies do not challenge shutdown techniques as the behavior of mechanical technologies (like HDD) does. Indeed, contrarily to HDD, SSD's drawbacks do not include limited number of on/off cycles (28), and can thus smoothly accommodate with any server shutdown policy. Finally, hybrid systems like solid state hybrid drive (SSHD) combine both technologies and consequently, they exhibit a warranted number of load/unload cycles identical to HDD technologies alone (29).

4.5 | Simulation Tool

In order to exploit the workload traces described in Section 4.1 and to combine them with our measurements on the Grid'5000 testbed (Section 4.2), we have designed a specific simulation framework. This simulation tool has been developed in Java and replays the studied workloads with the different shutdown policies that are explored in this paper. Consequently, the simulation tool has three inputs: the trace to replay, the calibration measurements (on-off sequences monitored on a real platform), and the shutdown policy to apply.

During the replay, on every idle period encountered for each machine, the simulation tool determines if the chosen policy would have switched the machine off and then on, instead of keeping it idle. As we did not have physical access to all the machines appearing in the obtained workload traces for monitoring the on-off sequences, we assume that all the machines are homogeneous and that they follow the calibrated sequences detailed in Tables 1 and 2 . Since our aim is to estimate the potential energy gains of shutdown techniques without impacting the users' performances, the simulated replays do not modify the working sequences: computing jobs are not delayed due to on-off sequences. Indeed, if an idle period of time lasts less than T_s seconds (as defined in Section 3.1), it is not considered for a shutdown sequence.

5 | IMPACT OF SHUTDOWN POLICIES

This section explores the simulation results of the shutdown policies with the various hardware calibrations and the workload traces described in Section 4. For every trace replay, the nodes are assumed to be homogeneous. Thus, every node of the trace is respecting the configuration of one of the calibrated nodes for each run. This assumption situates our results in the context of an homogeneous cluster for clarity's sake: in case of opposite effects on different server architectures, results would not be meaningful.

We first examine the case of current architectures based on the calibration made on the Grid'5000 nodes and described in Table 1 . While policy P1 always performs exact prediction of the future workload in order to adequately switch on and off the nodes, policy P2 does not attempt to foresee the future and switches off a node as soon as it is unemployed. The ideal case assumes that state transitions have no cost in terms of energy and time, but switched off nodes are still consuming a bit ($P_{off} \neq 0$), so the energy gains of the ideal case is not 100%.

Table 3 shows the percentage of energy that could be saved during idle periods with each policy compared to the energy consumed if nodes are never switched off. The last two columns present the average number of On-off cycles per node for the entire duration of the workload (respectively 6 years and 15 months for the two workload traces).

The results show that by turning off nodes, even when considering On-Off and Off-On costs, consequent energy gains can be made on real platforms. In the most unfavorable configuration (ie. Orion configuration), by using shutdown techniques, we can theoretically save up to 86% of the energy consumed while being in idle state. In the case of Grid'5000 trace, this percentage represents around 706,000 kWh for the 6 years, so roughly a cost of 70,600 euros (at a cost of 0.10 euros per kWh). For the E-Biothon trace, we can also save up to 86% of the energy consumed in the idle case, this represents 109,000 kWh for 15 months, roughly 10,900 euros of loss to keep servers idle.

The number of On-Off cycles per node reaches at the maximum $5,690$ for the 6-year Grid'5000 traces, so 2.59 per day, far less than the 50,000 start/stop cycles typically allowed by HDD manufacturers during their 5-year lifetime under warranty (24, 25). This clearly states that even aggressive shutdown policies have no impact on disk lifetime.

It is worth noticing that significant energy gains can be performed for both traces even though they present completely different use cases. In particular, the E-Biothon trace comes from an operational bioinformatics supercomputer and although energy savings are smaller than for the Grid'5000 trace in comparison with the infrastructure size, they are still not negligible, representing around 73,680 kWh per year for the Orion case (most unfavorable case) with a basic shutdown policy like P2 (without prediction algorithm).

TABLE 3 Energy gains on idle periods and number of on-off cycles per node for current servers

Calibration	% Energy saved on idle periods			# On-Off cycles per node	
	P1	P2	Ideal	P1	P2
Grid'5000 trace, 6 years, 149 nodes on average					
Orion	85.87%	85.59%	86.29%	3,080	5,690
Taurus	90.56%	90.22%	91.05%	2,980	5,690
Paravance	96.66%	96.46%	97.00%	3,333	5,690
E-Biothon trace, 15 months, 4096 nodes					
Orion	85.18%	84.56%	86.29%	33	70
Taurus	89.83%	89.07%	91.05%	33	70
Paravance	96.03%	95.61%	97.00%	38	70

The energy saved with policies P1 and P2 are very close to the ideal case (around 2% difference in the worst case). Even without knowledge about the future (policy P2), energy gains are quite similar. This means that even simple shutdown policies – not including workload predictions – can save consequent amounts of energy, close to the optimal bound in the context of the studied types of workload. These results show that the energy gains of P1 and P2 is too close (for Orion 0.28% of difference between the policies, roughly 2,000kWh over 6 years) to justify the elaboration of a prediction algorithm: such a complex algorithm to design would only bring negligible benefits. Such a conclusion can differ in the context of other types of workloads with small and frequent idle periods for instance.

6 | IMPACT OF FUTURE ENERGY-AWARE HARDWARE

6.1 | Experiments on future architectures with improved shutdown modes

After this analysis on current hardware, we study the impact of shutdown techniques on envisioned future architectures: regular nodes with an S3 mode (Suspend-to-RAM) and power-proportional nodes. For the S3 mode, it was not available on the Grid'5000 servers used for our calibration measurements. However, one can assume that when this technology will become more mature and used in hardware composing datacenters, it could present an appealing trade-off between energy consumption (for switching off nodes) and reactivity (for their short switching time T_{OnOff} and T_{OffOn}).

TABLE 4 Assumed energy calibration on envisioned nodes with S3 mode

Parameters	Values
E_{OffOn} (Joules)	2,300
E_{OnOff} (Joules)	2,300
T_{OffOn} (seconds)	10
T_{OnOff} (seconds)	10
P_{idle} (Watts)	135
P_{off} (Watts)	37
T_s (seconds)	20

After discussing with people from the Leibniz Supercomputing Centre operating the SuperMUC HPC system (30) on which S3 is available, we get quantitative indications stating that, the power consumption on S3 mode is about twice bigger than when regularly switched off, and the On-Off and Off-On sequences are close in terms of duration. So, based on an Orion calibration from our measurements (as presented in Table 1), we assume that an envisioned node with S3 mode would present the energy calibration parameters shown in Table 4 .

Table 5 presents the comparison of the energy gains between this S3 mode and a regular shutdown (S5) for the Orion case as shown in previous results presented in Table 3 . Results indicate that S5 mode allows for more energy savings than S3 mode on these traces. Indeed, idle periods are long enough to easily compensate for the energy and time costs of switching between states. However, the consumption while in S3 mode (P_{off})

is, in this case, too high for competing with the energy saving percentage reached with a regular shutdown. For the S3 mode to be beneficial for workloads with consequent idle periods, it is thus required to diminish its energy consumption (P_{off}) rather than reducing the switching costs (and thus T_s).

TABLE 5 Energy gains on idle periods and number of on-off cycles per node with an envisioned S3 mode

	% Energy saved on idle periods			# On-Off cycles per node	
Calibration	P1	P2	Ideal	P1	P2
Grid'5000 trace, 6 years, 149 nodes on average					
Orion	85.87%	85.59%	86.29%	3,080	5,690
S3	72.51%	72.48%	72.59%	3,343	5,690
E-Biothon trace, 15 months, 4096 nodes					
Orion	85.18%	84.56%	86.29%	33	70
S3	72.31%	72.26%	72.59%	52	70

6.2 | Experiments on future energy-proportional architectures

Concerning power-proportional nodes, results are provided by Table 6 based on the calibration values and configurations exposed in Section 4.3. As expected, for both policies, when only one component over the three units composing the processing node can be switched off (cases OnlyL, OnlyM and OnlyB), it consumes more than when the three can (AtomicBML and FlexibleBML).

Moreover, switching off only the Little or the Medium components result in little energy savings (less than 9%). This explains that FlexibleBML – able to switch off the three components independently or together – brings minor improvements compared to AtomicBML, where the three components are always switched jointly (0.6% difference on policy P1). For policy P2 and configuration FlexibleBML, it gives the same results as configuration AtomicBML because this policy automatically switches down all the components whenever possible, so it produces the same behavior as AtomicBML in this case.

TABLE 6 Shutdown impacts with an energy-proportional architecture

	% Energy saved on idle periods			# On-Off cycles per node	
Calibration	P1	P2	Ideal	P1	P2
Grid'5000 trace, 6 years, 149 nodes on average					
AtomicBML	77.66%	77.51%	77.91%	3,495	5,690
OnlyL	2.00%	2.00%	2.007%	5,690	5,690
OnlyM	8.93%	8.93%	8.941%	5,690	5,690
OnlyB	72.19%	72.05%	72.44%	3,511	5,690
FlexibleBML	77.72%		77.91%	5,690	
E-Biothon trace, 15 months, 4096 nodes					
AtomicBML	77.22%	76.93%	77.91%	42	70
OnlyL	2.00%	2.00%	2.007%	70	70
OnlyM	8.93%	8.93%	8.941%	70	70
OnlyB	71.78%	71.50%	72.44%	42	70
FlexibleBML	77.72%	76.93%	77.91%	70	70

Similarly to previous results, we observe that policy P1 and P2 give comparable results (0.79% of difference at maximum), and they are close to ideal case (0.98% at most). Designing an accurate workload prediction algorithm has therefore little interest for energy saving purpose in the studied context. In the same way as previous results also, the number of On-Off per cycle is small enough not to modify the hardware life expectancy.

7 | CONCLUSION AND FUTURE WORK

The energy-efficiency of servers is increasing with Moore's law. Yet, due to an increased demand for Internet-based services, the energy consumption of large-scale systems keeps growing and is becoming more and more a worrying concern. Although shutdown techniques are available to reduce the overall energy consumption during idle periods, they are rarely employed because of their supposed impact on hardware.

Simulation results combining real workload traces and energy calibration measurements conducted in this paper allow us to draw several conclusions:

- Shutdown techniques can save – even in production data centers – important amounts of energy otherwise wasted during idle periods and this conclusion remains true for envisioned future hardware with power-proportional processing units.
- Even aggressive shutdown policies have no negative impact on disk lifetime.
- Reducing the consumption while in Off state has a greater impact on energy savings than reducing the switching energy and time costs between On and Off states. For this reason, S3 (Suspend-to-RAM) and S4 (Suspend-to-Disk) states are currently not beneficial in terms of energy consumption.
- Workload prediction can improve the reactivity of the system implementing shutdown techniques. Nevertheless, for the types of workload studied in this paper, we underline the fact that prediction is not necessary to save important amounts of energy.

As stated by (10), switching on and off large scale infrastructures can be a real challenge due to several constraints: temperature, power capping, renewable energy provision, etc. Our future work includes an integration of failure models when resuming from Off state in order to study the impact of bad resuming behavior. We also plan to evaluate other shutdown policies which are applied in current data centers like switching nodes by portions of the total number to control the impact on data center cooling system.

ACKNOWLEDGMENTS

Experiments presented in this paper were carried out using the Grid'5000 experimental testbed, being developed under the INRIA ALADDIN development action with support from CNRS, RENATER and several Universities as well as other funding bodies (see https://www.grid5000.fr).

This work is integrated and supported by the ELCI project, a French FSN ("Fond pour la Société Numérique") project that associates academic and industrial partners to design and provide software environment for very high performance computing.

References

[1] ABB Infographic. Powering the cloud http://www.abb.com/cawp/seitp202/7dc34cb95e084519c125788e00575508.aspx2015.

[2] Orgerie Anne-Cécile, Assunção Marcos, Lefèvre Laurent. A Survey on Techniques for Improving the Energy Efficiency of Large-scale Distributed Systems. *ACM Computing Survey.* 2014;46(4):47:1–47:31.

[3] Orgerie Anne-Cécile, Lefèvre Laurent. ERIDIS: Energy-Efficient Reservation Infrastructure for Large-scale Distributed Systems. *Parallel Processing Letters.* 2011;21(02):133-154.

[4] Lovász Gergo, Niedermeier Florian, Meer Hermann. Performance tradeoffs of energy-aware virtual machine consolidation. *Cluster Computing.* 2013;16(3):481-496.

[5] Srikantaiah Shekhar, Kansal Aman, Zhao Feng. Energy Aware Consolidation for Cloud Computing. In: *USENIX Conference on Power Aware Computing and Systems (HotPower)* :1–5; 2008.

[6] Gmach Daniel, Rolia Jerry, Cherkasova Ludmila, Kemper Alfons. Resource Pool Management: Reactive Versus Proactive or Let's Be Friends. *Computer Networks.* 2009;53(17):2905–2922.

[7] Rais Issam, Orgerie Anne-Cécile, Quinson Martin. Impact of Shutdown Techniques for Energy-Efficient Cloud Data Centers. In: *ICA3PP: International Conference on Algorithms and Architectures for Parallel Processing*:203-210; 2016; Granada, Spain.

[8] Chase Jeffrey S., Anderson Darrell C., Thakar Prachi N., Vahdat Amin M., Doyle Ronald P.. Managing Energy and Server Resources in Hosting Centers. In: *ACM Symposium on Operating Systems Principles (SOSP)* :103–116; 2001.

[9] Pinheiro Eduardo, Bianchini Ricardo, Carrera Enrique V., Heath Taliver. Load Balancing and Unbalancing for Power and Performance in Cluster-Based Systems. In: *Workshop on Compilers and Operating Systems for Low Power* :182-195; 2001.

[10] Benoit Anne, Lefèvre Laurent, Orgerie Anne-Cécile, Rais Issam. Reducing the energy consumption of large scale computing systems through combined shutdown policies with multiple constraints. *International Journal of High Performance Computing Applications.* 2017;.

[11] Demaine Erik D., Ghodsi Mohammad, Hajiaghayi Mohammad Taghi, Sayedi-Roshkhar Amin S., Zadimoghaddam Morteza. Scheduling to Minimize Gaps and Power Consumption. In: *Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA)* :46–54; 2007.

[12] Gandhi Anshul, Gupta Varun, Harchol-Balter Mor, Kozuch Michael A.. Optimality analysis of energy-performance trade-off for server farm management. *Performance Evaluation.* 2010;67(11):1155–1171.

[13] Lin Minghong, Wierman Adam, Andrew Lachlan L. H., Thereska Eno. Dynamic Right-sizing for Power-proportional Data Centers. *IEEE/ACM Transactions on Networking (TON).* 2013;21(5):1378–1391.

[14] Yoo Andy B., Jette Morris A., Grondona Mark. SLURM: Simple Linux Utility for Resource Management. In: *International Workshop Job Scheduling Strategies for Parallel Processing (JSSPP)* :44–60; 2003.

[15] Pernici Barbara, Cappiello Cinzia, Fugini MariaGrazia, et al. Setting Energy Efficiency Goals in Data Centers: The GAMES Approach. In: Huusko Jyrki, Meer Hermann, Klingert Sonja, Somov Andrey, eds. *Energy Efficient Data Centers*, Lecture Notes in Computer Science, vol. 7396: Springer 2012 (pp. 1-12).

[16] Orgerie Anne-Cécile, Lefèvre Laurent, Gelas Jean-Patrick. Save Watts in Your Grid: Green Strategies for Energy-Aware Framework in Large Scale Distributed Systems. In: *IEEE International Conference on Parallel and Distributed Systems (ICPADS)* :171-178; 2008.

[17] Zhang W., Wen Y., Wong Y. Wah, Toh K. Chuan, Chen C. H.. Towards Joint Optimization Over ICT and Cooling Systems in Data Centre: A Survey. *IEEE Communications Surveys Tutorials.* 2016;18(3):1596-1616.

[18] Zhan Xin, Reda S.. Techniques for energy-efficient power budgeting in data centers. In: *ACM/EDAC/IEEE Design Automation Conference (DAC)* :1-7; 2013.

[19] Balouek Daniel, Carpen Amarie Alexandra, Charrier Ghislain, et al. Adding Virtualization Capabilities to the Grid'5000 Testbed. In: Ivanov Ivan I., Sinderen Marten, Leymann Frank, Shan Tony, eds. *Cloud Computing and Services Science*, Communications in Computer and Information Science, vol. 367: Springer International Publishing 2013 (pp. 3-20).

[20] Daydé Michel, Depardon Benjamin, Franc Alain, et al. E-Biothon: an experimental platform for BioInformatics. In: *International Conference on Computer Science and Information Technologies (CSIT)* :1–4; 2015.

[21] Jeff Brian. Big.LITTLE system architecture from ARM: saving power through heterogeneous multiprocessing and task context migration. In: *Annual Design Automation Conference (DAC)* :1143–1146; 2012.

[22] Villebonnet Violaine, Costa Georges Da, Lefèvre Laurent, Pierson Jean-Marc, Stolf Patricia. "Big, Medium, Little": Reaching Energy Proportionality with Heterogeneous Computing Scheduler. *Parallel Processing Letters.* 2015;25(3).

[23] Villebonnet Violaine, Costa Georges Da, Lefèvre Laurent, Pierson Jean-Marc, Stolf Patricia. Towards Generalizing "Big Little" for Energy Proportional HPC and Cloud Infrastructures. In: *IEEE International Conference on Big Data and Cloud Computing (BDCloud)* :703–710; 2014.

[24] Seagate . Desktop HDD specification sheet http://www.seagate.com/staticfiles/docs/pdf/datasheet/disc/desktop-hdd-data-sheet-ds1770-1-1212us.pdf2012.

[25] Seagate . NAS HDD specification sheet http://www.seagate.com/www-content/product-content/nas-fam/nas-hdd/_shared/docs/nas-hdd-8tb-ds1789-5-1510DS1789-5-1510US-en_US.pdf2015.

[26] He S., Wang Y., Sun X. H., Huang C., Xu C.. Heterogeneity-Aware Collective I/O for Parallel I/O Systems with Hybrid HDD/SSD Servers. *IEEE Transactions on Computers.* 2017;66(6):1091-1098.

[27] Yin S., Xiao Z., Li K., et al. RESS: A Reliable Energy-Efficient Storage System. In: *IEEE International Conference on Parallel and Distributed Systems (ICPADS)* :1193-1198; 2016.

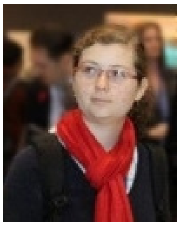
[28] Seagate . 1200 SSD specification sheet https://www.seagate.com/files/www-content/product-content/ssd-fam/1200-ssd/en-gb/docs/1200-ssd-ds1781-4-1310us.pdf2015.

[29] Seagate . Desktop SSHD specification sheet https://www.seagate.com/www-content/product-content/barracuda-fam/desktop-sshd/en-us/docs/desktop-sshd-data-sheet-ds1788-2-1308us.pdf2015.

[30] Auweter Axel, Bode Arndt, Brehm Matthias, et al. A Case Study of Energy Aware Scheduling on SuperMUC". In: *International Conference on Supercomputing (ISC)* :394–409; 2014.

AUTHOR BIOGRAPHY

Issam Raïs is a PhD Student in the Algorithms and Software Architectures for Distributed and HPC Platforms (Avalon) team from the LIP laboratory in Ecole Normale Superieure of Lyon, France. His interests include energy efficiency for high performance computing systems.



Anne-Cécile Orgerie received her PhD. degree in Computer Science from École Normale Supérieure de Lyon (France) in September 2011. Then she spent one year as a postdoc at the Department of Electrical and Electronic Engineering at the University of Melbourne (Australia) on the PetaFlow project (French-Japanese project). She has been a full time researcher at CNRS in the IRISA laboratory (Rennes, France) since October 2012. Her research interests focus on energy efficiency, cloud computing, green networking and distributed systems.

Martin Quinson is a Professor at Ecole Normale Supérieure de Rennes since 2015. Before he was an assistant professor at the Université de Lorraine. His research is on experimentation methodologies, both through the simulation of distributed applications and through the formal assessment of distributed algorithms. He obtained his M.S. and Ph.D. from the Ecole Normale Supérieure de Lyon respectively in 2000 and 2003, and his Habilitation Thesis in 2013 from the Université de Lorraine, France.

Laurent Lefèvre is a permanent researcher in computer science at Inria (the French Institute for Research in Computer Science and Control). He is a member of the Algorithms and Software Architectures for Distributed and HPC Platforms (Avalon) team from the LIP laboratory in Ecole Normale Superieure of Lyon, France. He has organized several conferences in high performance networking and computing and he has been member of several program committees. He has co-authored more than 100 papers published in refereed journals and conference proceedings. His interests include energy efficiency in large-scale distributed systems, high performance computing, distributed computing and networking, high performance networks protocols and services.

How cite this article: Issam Rais, Anne-Cécile Orgerie, Martin Quinson, and Laurent Lefèvre (2017), Impact of Shutdown Techniques for Energy-Efficient Cloud Data Centers, *Concurrency & Computation: Practice & Experience, 2017;XX:X–X.*