IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 18, 2020, accepted February 29, 2020, date of publication March 4, 2020, date of current version March 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978306

Limiting Global Warming by Improving Data-Centre Software

DAMIÁN FERNÁNDEZ-CERERO[1], ALEJANDRO FERNÁNDEZ-MONTES[1], AND AGNIESZKA JAKÓBIK[2]

[1]Department of Languages and Systems, University of Seville, 41012 Seville, Spain
[2]Department of Computer Science, Cracow University of Technology, 31-155 Cracow, Poland

Corresponding author: Damián Fernández-Cerero (damiancerero@us.es)

This work was supported by the FEDER/Spanish Ministry of Science and Innovation, National Research Agency, under Project RTI2018-098062-A-I00.

ABSTRACT Carbon emissions, greenhouse gases and pollution in general are usually related to traditional factories, so the most modern computing *factories* have gone unnoticed for the general-public opinion. We empirically show through extensive and realistic simulation that: 1) energy consumption, and consequently CO_2 emissions, could be reduced from ∼15% to ∼60% if the correct energy-efficiency policies are applied; and 2) such energy-consumption reduction can be achieved without negatively impacting the correct operation of these infrastructures. To this end, this work is focused on the proposal and analysis of a set of energy-efficiency policies which are applied to traditional and hyper-scale data centres, as well as numerous operation environments, including: 1) the top resource managers used in industry; 2) eight energy-efficiency policies, including aggressive, fine-tuned and adaptive models; and 3) three types of workload-arrival patterns. Finally, we present a realistic analysis of the environmental impact of the application of such energy-efficiency policies on USA data centres. The presented results estimate that 11.5 million of tons of CO_2 could be saved, which is equivalent to the removal of 4.79 million of combustion cars, that is, the total car fleet of countries such as Portugal, Austria and Sweden.

INDEX TERMS Energy efficiency, data centres, scheduling.

I. INTRODUCTION

One degree Celsius. It is the estimated global temperature change induced by human activities. 2006 - 2015 is assessed to be 0.87°C increased global temperature [2]. *Every* bit of help reducing carbon footprint and greenhouse gases emissions is needed. Every effort must be done.

And it is *urgent*.

Currently, data centres represent some of the greediest industries worldwide in terms of energy consumption. The ever-increasing utilisation of cloud services is increasing the demand of these facilities. According to the latest studies, up to 2% [3] of global energy is consumed by data centres. Some of them have adopted green energy-generation and consumption strategies, however, the majority of their energy consumption is provided by the traditional power grid, which leads to non-green energy generation. According to [4] around 60% of the energy generation comes from coal and natural gas sources, thus CO^2 is still wildly emitted to the atmosphere.

Data-centre energy consumption may be categorised by its consumer, from computational resources to storage and networking. In addition, the more computational work the servers have to perform, the more heat is generated, and the more energy is consumed by the cooling systems.

Various strategies may be adopted to save energy in data centres, from industrial-oriented models to software-management solutions. Industrial-oriented approaches include improvements on data-centre cooling and temperature of operation [5], [6], hardware energy proportionality [7], [8], replacement of hardware components to incorporate more-efficient and less-consuming components [9], and the improvement of power-distribution systems [10].

The main software systems in charge of the deployment and execution of the incoming workload are known as resource managers. Thus, resource managers have to deal with the arrival of jobs, the status of the resources and the allocation of resources to such jobs. In order to reduce energy consumption, we analyse a wide range of power-off

The associate editor coordinating the review of this manuscript and approving it for publication was Yinliang Xu.

policies that dynamically and virtually change the number of data-centre resources, therefore adjusting the available resources for the current and near-future workload. These energy policies have been incorporated to the most popular resource-management software solutions, and simulation tools have been used in order to obtain results and to analyse them in consequence. Economic and environmental results, including CO^2 savings, are also detailed.

This work focuses on the reduction of the energy consumed by the workloads that are executed on data-centre computational resources from a software-governance point of view. The following novelties are therefore presented in this paper:

- Two new self-adaptive energy-efficiency policies based on the shut-down of underutilised servers: BF-size and BF-time.
- Performance and energy-efficiency analysis of the developed energy-efficiency policies in various kind of realistic data centres and workloads: traditional and hyper-scale data centres.
- Measurement, analysis and presentation of the results of the developed energy-efficiency policies in two-level and shared-state resource managers.
- Measurement of the environmental impact of such energy-efficiency policies at USA level.

Practically, this paper enables data-centre administrators to make decisions related to the application of energy-efficiency-related policies based on empirical information obtained through extensive and realistic simulation.

The paper is organised as follows: the related work is described in Section III. In Section II we introduce the foundations of the data-centre operation process, including an overview of the software that governs these infrastructures, and the workload to be executed. Section IV presents the experimentation design and the simulation tool employed. Finally, empirical results are presented and analysed in Section V, where we compare 18 different scenarios, considering two data-centre infrastructures (traditional and hyper-scale), three resource managers and three workload patterns. Various energy policies are applied to each scenario, and numerous results related to energy efficiency and Key Performance Indicators (KPIs) are presented. Moreover, we provide some results related to the environmental impact of these solutions if applied to USA data centres. Conclusions and future work are stated in Section VI.

II. DATA-CENTRE ARCHITECTURE

Data centres are very complex infrastructures that usually provide storage and computational resources to services and applications deployed by final users and operators. Therefore, software solutions have to manage large amounts of digital storage, servers (aka nodes) to perform computational tasks, both the external and internal networks for successful and low-latency communication, and environmental control for cooling and dehumidification.

Data centres may be categorised by its size and resource utilisation. As stated by Shehabi [3] we differentiate between 1) hyper-scale; and 2) traditional data centres. Hyper-scale data centres are usually composed of a large number of resources which operate at high utilisation rates to service cloud and big-data operations. Such data centres are usually operated by large companies such as Google, Microsoft, and Amazon. Traditional data centres are usually composed of a lower number of resources, whose utilisation rate are usually lower due to software and business limitations. Traditional data centres are usually related to private-cloud solutions owned by smaller organisations to support their internal business services.

Each data centre is managed and operated by a complex stack of software solutions. The particular software stack deployed on production depends on the data-centre administrator. This decision is usually related, in turn, to the purpose of the services provided by the data centre. In consequence, the software stacks deployed in data centres worldwide are numerous and very heterogeneous.

This being said, data centres usually equip a set of software solutions which provide common functionality for their correct operation and management. Among them: 1) resource managers; 2) distributed file systems; 3) network load balancing systems; and 4) monitoring and management systems.

A. RESOURCE MANAGERS

Resource managers are the main software tool that governs data-centre operations. It is in charge of the management, monitoring and deployment of jobs and their tasks by providing the following features: 1) management of the arrival and queuing of task; 2) selection of resources that meet the requirements; 3) perform the operations needed to actually deploy such tasks; and 4) monitoring the state of the resources and the life cycle of the deployed jobs and tasks.

One of the main characteristics of resource managers is their scheduling algorithms that carry out the selection of resources. These schedulers are a critic part of the overall resource governance. In this context, several alternatives have been proposed for the architecture of the cloud-computing resource managers. At the highest level we can classify them as: a) Centralised; b) Distributed; and c) Hybrid.

On the one hand, centralised resource managers present a unique authority with full knowledge of all the operational status. Nowadays these are the most popular alternatives for data centres in production. They offer fairly good performance and are the most tested alternatives. In this work, we will propose energy-efficiency solutions applied to this group of resource managers.

On the other hand, we can find distributed resource managers, where no central authority is in charge of all the tasks, but some local instances work independently to manage and operate the resources and queuing/deployment of jobs. Scheduling decisions are usually sub-optimal, as each instance has little knowledge of the status the rest of the resources.

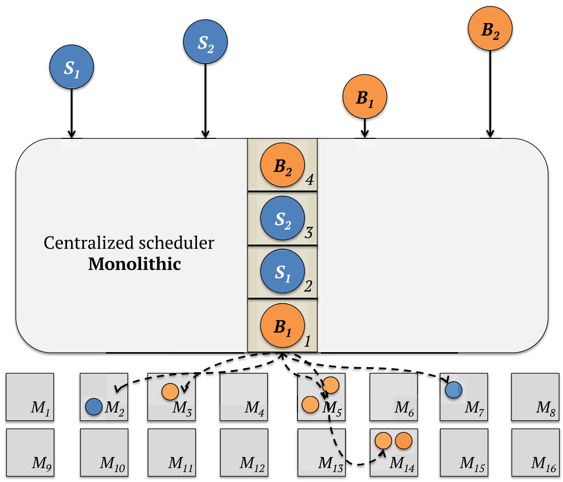

FIGURE 1. Monolithic centralised scheduling workflow.

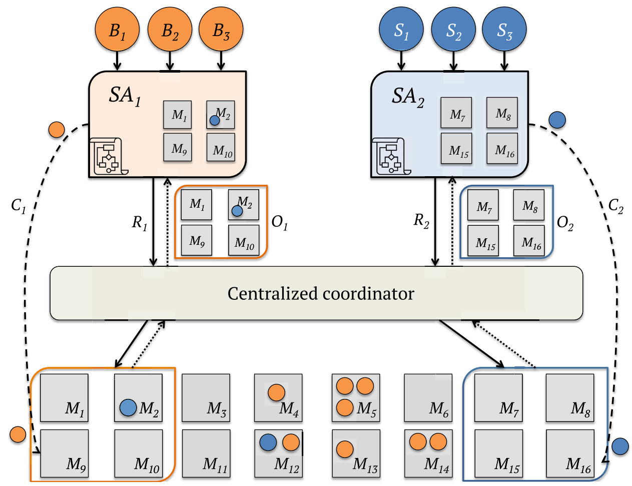

FIGURE 2. Two-level centralised workflow.

Finally, hybrid approaches join both alternatives in a hierarchical model. At the top, a central meta-manager interacts with the local job dispatchers. Such meta-manager organises the sets of jobs and resources whilst second-level managers control medium-sized cloud clusters. These local managers are not aware of the rest of the data centre. This approach results in a more scalable and fault-tolerant architecture while it keeps the advantages of centralised schemes.

The presented categorisation works at a very high level. These categories can be scattered in several subcategories. Thus, in this work, we analyse the following three most utilised centralised resource managers: a) monolithic schedulers; b) two-level schedulers; and c) shared-state schedulers.

The following figures illustrate each of these resource managers and use some abbreviations as follows: B - Short-running Batch task, S - Long-running Service task, M - Machine, O - Resource offer, SA - Scheduling Agent, R - Request of computational resources, C - Commit of scheduling operation, U - Cluster state Update.

1) MONOLITHIC CENTRALISED RESOURCE MANAGERS

Monolithic centralised manager is the prevalent model of data-centre resource managers. This approach [11] works fine when the arrival of jobs presents high frequency of long-lasting tasks jobs where the latency of the response is not critical [12]. It usually features high-quality and near to optimal scheduling decisions [13], [14]. Moreover, the Monolithic model tends to utilise data-center resources at higher rates [15], which is related to a more homogeneous performance, reliable behaviour, fair load balance [16], [17], and shortening of makespan [18]. The workflow of the aforementioned schedulers is depicted in Fig.~1

Notwithstanding, some data centres need the partition of jobs in order to perform fast-response operations. It implies that jobs are divided into a high number of smaller and shorter tasks. When this situation is present, monolithic resource managers may impose a bottleneck in this data-centre operation.

2) TWO-LEVEL CENTRALISED RESOURCE MANAGERS

New centralised scheduling approaches have been proposed and developed in order to avoid or reduce the aforementioned bottleneck. These alternatives aim their efforts in the parallelisation of the scheduling decisions. Two-level centralised resource managers, including Mesos [19], and YARN [20], employ a central coordination agent that prevents the parallel modification of the cluster resources when scheduling operations are to be made. These decisions are performed by several schedulers, usually packed with user application frameworks, such as MapReduce. Each framework incorporates scheduling algorithms responsible for the selection of resources on which jobs are deployed, as show in Figure 2. Therefore, each scheduler lacks the global cluster state and tasks requirements which may lead to sub-optimal scheduling decisions.

3) SHARED-STATE CENTRALISED RESOURCE MANAGERS

In contrast with the aforementioned approach, *Shared-state centralised resource managers*, such as Omega [18], follow a different approach. The centralised coordinator offers the whole cluster state to all schedulers and does not block the cluster on each scheduling decision. Consequently, schedulers can work concurrently, employing a not-updated copy of the data-centre state. When a scheduler makes a decision, commits the scheduling transaction to the centralised scheduler. A conflict may occur as the result of the transaction, as the stale copy could be outdated. In this case, the coordinator updates the not-updated copies of the data-centre state which are used by the schedulers. Finally, the scheduling operation is retried, as shown in Figure 3.

B. DATA-CENTRE WORKLOADS

In current cloud-computing environments, the processed workload presents a high degree of heterogeneity [21], [22]. Workloads are usually divided depending on the duration and goal of their jobs into batch or service workloads:

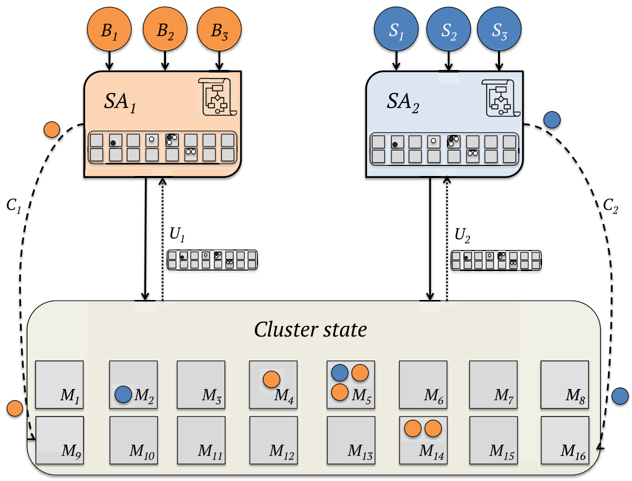

FIGURE 3. **Shared-state centralised scheduling workflow.**

- **Batch** jobs are the ones that perform a clearly defined computation task, which leads to a fixed end time when the computation finishes (e.g. Map-reduce jobs).
- **Service** jobs usually represent infrastructure services and processes which have no determined end, such as distributed file system instances, mail and web servers.

Batch jobs usually comprise 90% of total jobs scheduled in the data centre, and service jobs are usually 10% of total jobs scheduled. In contrast, both batch and service jobs consume approximately the same amount of data-centre resources, 45% and 55%, respectively [23]–[25]. It must be kept in mind that the sub-optimal scheduling operations of distributed schedulers may cause a severe performance degradation on complex and long-running jobs, whose key performance indicators may reflect a worse performance than in overwhelmed centralised managers.

III. ENERGY EFFICIENCY IN DATA CENTRES

Besides the efforts made in the various hardware, electric and industrial areas, numerous software-based energy-efficiency models have been also proposed. Such models may be classified in: 1) virtual machines consolidation, migration and scaling; 2) energy-aware scheduling algorithms; and 3) power-off policies which shut down idle nodes according to the workload patterns.

In addition, different techniques of energy conservation such as VM consolidation and migration have been developed in literature. In [26]–[28], the authors propose several resource-management models for virtualised data centres in order to lower energy consumption: 1) allocation and migration of VMs depending on CPU usage; 2) taking into account SLAs restrictions; and 3) adaptive heuristics.

A Bayesian Belief Network-based algorithm whose objective is the allocation and migration of VMs is presented in [29], which takes into account the data gathered during the execution of the tasks.

In [30] multi-level Join VM Placement and Migration (MJPM) algorithms based on the relaxed convex optimization framework (taking into account energy cost of VMs migration) to approximate the optimal solution are proposed. Finally, a Multi-objective Genetic Algorithm is proposed by [31] for the dynamic prediction and allocation of resources.

The authors propose an energy-aware scheduling policy based on Dynamic Voltage and Frequency Scaling (DVFS) in [32]. Moreover, various approaches look for the reduction of the energy consumption by applying energy-proportionality models based on power-proportional distributed file systems in [33]–[35] which generally aim to switch storage-servers off when the replicated data is not needed.

In [36], the authors propose a green-scheduling algorithm based on neural networks, whose objective is the forecast of workload demand to switch off idle resources. Several experiments are performed by simulating a 512-nodes data centre with homogeneous workload which follows a day/night pattern. In [37], the authors describe two energy-aware heuristics that aim to maximize resource utilization. Moreover, heuristic rules and resource allocation techniques are proposed to minimise a multi-objective function, taking into account the energy-consumption and execution time in [38]. The authors in [39] describe an energy manager that relies on day/night workload patterns to aggregate jobs and therefore switch idle nodes off, while keeping a security margin to minimise unexpected workload peaks.

This work can be classified within the power-off policies that shut-down servers. To this end we describe 8 different power-off policies and compare them with the current situation, represented by a non-switching-off policy. The comparison includes performance and energy-consumption metrics. These energy-saving policies are applied at the resource-manager layer, making it possible to be put in production on the vast majority of data-centre models.

Moreover, we compare the behaviour of the data centre when these policies are applied to various scenarios, specifically:

1) Two data-centre workload utilisation levels which represent the usage of hyper-scale data centres and the big and medium traditional data centres;
2) Three data-centre workload patterns that correspond to the most popular day/night patterns that follow real cluster traces; and
3) The three centralised resource managers that match the most popular resources managers found in real data centres in production.

This paper extends our previous work [40]–[45] and the related work found in the literature presented in this section by:

1) Presenting two new self-adaptive energy-efficiency policies based on the shut-down of underutilised servers: BF-size and BF-time.
2) Applying the developed energy-efficiency policies to various kind of realistic data centres and workloads: traditional and hyper-scale data centres.

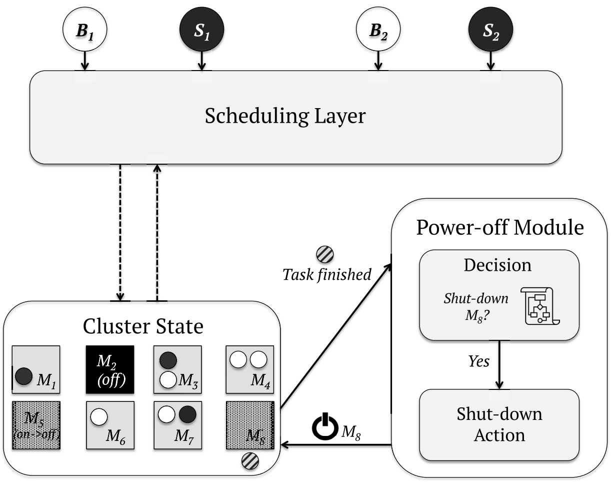

FIGURE 4. **Power-off module architecture.**

3) Applying, measuring and analysing the results of the developed energy-efficiency policies in two-level and shared-state resource managers.
4) Measuring the environmental impact of such energy-efficiency policies at USA level.

A. POWER-OFF POLICIES

In this work, we propose the shut-down of machines to dynamically change the available capacity of the data centre to better fit the resources needed for the execution of the workload at a particular time. The shut-down of machines is led by the power-off policies. These policies determine whether or not a machine should be shut down and are responsible for the activation of the shut-down process. The workflow of this process is illustrated in Figure 4. The developed energy-efficiency policies inspect various data-centre and resources data to make the shut-down decisions. Such decisions are made after task completion, when computational resources are released. In our work,

The following shut-down energy-efficiency policies have been developed:

- **Keep on**: This shut-down policy prevents any resource from being shut-down. Therefore, this policy reflects the current behaviour of data centres.
- **Aggressive**: This shut-down policy will try to power off all the free resources after tasks are completed and resources released.
- **Minimum free-capacity margin**: This shut-down decision policy ensures that a given percentage of resources μ is available, at least, ready to accept incoming workload to avoid performance degradation.
- **Random**: This policy shuts down or leaves resources idle by following a Bernoulli distribution of 0.5. Results could be useful in order to make comparisons and check the accuracy made by the probabilistic policies presented below.
- **Exponential**: Exponential distribution, $Exp(\lambda)$, has been widely used to describe the time between events of a Poisson process. In the context of data centres, the arrival the new jobs are usually modelled following the Queue Theory models. Therefore, this policy offers a prediction of the arrival of new jobs. In order to compute the λ parameter, only the most recent jobs are taken into account, based on a parameter denoted as *Window size*. Every time we need to make a switch-off decision, the mean inter-arrival time between those last jobs is computed, denoted by δ, and $\lambda = 1/\delta$ based on the method of maximum likelihood. Then, we compute the probability of the arrival of a new job in a given time by means of the cumulative density function (cdf), as $cdf(T_s)^1 = 1 - e^{-T_s/\delta}$. If the probability is greater or lesser than a given decision threshold μ, the resources are left idle or switched-off, respectively.
- **Gamma**: We use Gamma distributions, usually denoted as $\Gamma(\alpha, \beta)$, as a probability model for the occurrence of multiple events in a specific time interval. In this energy-efficiency policy, we follow the hypothesis that the arrival of new jobs follows a Gamma distribution. Thus, this policy provides with the probability of the arrival of enough new jobs to oversubscribe the current available resources in a given period, so a smart shut-down decision can be made. To this end, every time a shut-down operation is commanded, the energy-efficiency policy computes the availability of memory and cpu of the current idle resources, the memory and cpu needed to serve the last jobs, the mean inter-arrival time among these last jobs and the ration between cpu, α_{cpu}, and memory α_{mem} available versus the cpu and memory needed by such last jobs. Finally, it computes a new Gamma distribution as the cumulative density function (cdf) with:

$$cdf(T_s) = \frac{\gamma(\alpha, \beta x)}{\Gamma(\alpha)}$$

Hence, if the resulting probability is greater or lesser than a given a threshold μ, the resources are left idle or switched-off respectively.
- **BF-size**: This policy tries to overcome the difficulties that the aforementioned developed policies present, i.e. the need of a fine-tuned parameterisation. To this end, we consider a window size of a given number of last jobs. Then we compute n predictions and if the average of these predictions is greater or lesser than a given threshold, then we leave resources idle or shut them down respectively. Each of the aforementioned predictions is obtained by generating a new virtual future workload that follows an exponential distribution modelled after the behaviour of the last window of jobs (interarrival, cpu and memory consumption).
- **BF-time** This policy follows the same pattern, but the window size is not computed from a given number of last jobs, but from the past jobs from a given time frame.

[1] T_S is defined *as the minimum time that ensures energy saving if a resource is switched off between two jobs* [46]

From that, it also computes n predictions as previously stated.

IV. EXPERIMENTATION

In this work, we analyse the impact of the previously explained energy-efficiency policies in the two main families of data centres, that are, ordinary and hyper-scale data centres.

Ordinary data centres, which represent traditional and less-efficient data centres, utilise 20% of resources in average, while flagship hyper-scale data centres are considered to achieve a utilisation rate of 50% [3].

Moreover, we analyse the impact of the centralised resource managers explained in Section II-A on the behaviour of both types of data centres.

In addition, we consider three kinds of workloads:

1) Ordinary workload, which utilises an Exponential distribution for the generation of the job arrival times;
2) Extreme workload, whose job inter-arrival time is generated by the means of a Weibull $\alpha 0.5$ distribution; and
3) Very extreme workload, which generates the job inter-arrival time by using a Weibull $\alpha 0.3$ distribution.

The Exponential distribution, as a particular case of the Gamma distribution, is commonly employed in queue-theory environments to model the arrival of tasks in a given period. The Weibull distribution is used as one of the main extreme-value distributions, which increase the probability of extreme arrival of tasks. The Exponential distribution is a particular case of Weibull distribution when $\alpha = 1$.

We have performed numerous simulations of 7 days of operating time, and similar synthetic workloads for every experiment, running 10 iterations for each parameterisation. The average of the provided results is shown. We evaluated the homogeneity of the results within populations by means of the t-student test. Moreover, these workloads follow the trends present in large companies such as Google [47], and Alibaba [48] data centres. To this end, two kind of workloads sharing the same data-centre resources are generated based on statistical distributions, as explained in subsection II-B:

- Batch workloads, (jobs which compute a task and then finish; e.g. MapReduce jobs). In these experiments, Batch workloads account for ~90% of deployed jobs (~26,000), consuming ~60% of the data-centre resources. Each Batch job is composed of an average of 180 tasks, which consume 0.3 CPU cores and 200 Mb of RAM for 180 seconds.
- Service workloads, (applications, services and frameworks with no fixed end; e.g. virtual machines). Service workloads account for ~10% of deployed jobs (approximately 2,500), which consume ~40% of the resources of the data centre. Each Service job is composed of an average 30 tasks, which consume 0.5 CPU cores and 700 Mb of RAM for 1000 seconds.

These workloads are executed in data centres of 2,000 machines. In our simulation experiments, ordinary data centres utilise 20% of the resources in average, whilst hyper-scale data centres occupy, in average, approximately 50% of the resources. Data-centre machines are considered to be homogeneous in terms of performance and energy consumption and are equipped with 8 cores and 16GB of RAM.

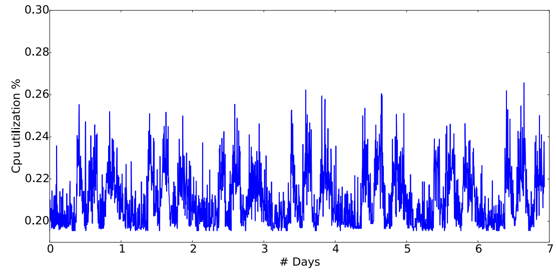

FIGURE 5. Example CPU consumption pattern of ordinary workloads.

A. WORKLOAD MODEL

A workload is a set of jobs that has to be deployed and executed by the data-centre resources (see Section II-B). Therefore, workloads are composed of jobs $\mathcal{W} = \{J_j\}_{j=1}^{n}$, and jobs are composed of tasks $\mathcal{T}_j = \{t_{ji}\}_{i=1}^{n_j}$. Thus, jobs are modelled by the following attributes:

- **Inter-arrival time** X_j represents the time between two consecutive jobs J_j and J_{j-1}. Therefore, it also influences the amount of jobs executed in a specific time window. The inter-arrival time between two *Batch* jobs is usually shorter than that of two *Service* jobs. According to the queue theory, the inter-arrival time usually follows an exponential distribution. Notwithstanding, other distributions, such as the extreme-value Weibull distribution may be employed.
- **Number of tasks** $n_j \sim Exp(\lambda_t)$ represents the number of tasks that comprise a job. The number of tasks of a particular job J_j is generated by following an Exponential distribution with a given mean value of $1/\lambda_t$.
- **Job duration** $d_j \sim Exp(\lambda_d)$ represents the period of time a given job J_j consumes resources in the data centre. The duration of all tasks of a particular job J_j is generated by the means of the Exponential distribution with the given expected value $1/\lambda_d$.
- **Resource usage** is the amount of CPU K_{CPU} and RAM K_{RAM} that all task of each particular job in a workload consumes.

B. WORKLOAD GENERATION MODELS

All the workloads used for these simulations present a day/night pattern. The following Figures 5, 6, 7 represent such day/night patterns in terms of resource consumption for ordinary, extreme and very extreme workloads generation models respectively.

C. PERFORMANCE MODEL

In order to evaluate and analyse the performance of the data centres, the behaviour of executed jobs is studied,

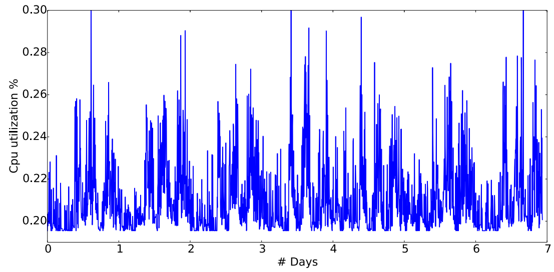

FIGURE 6. Example CPU consumption pattern of extreme workloads.

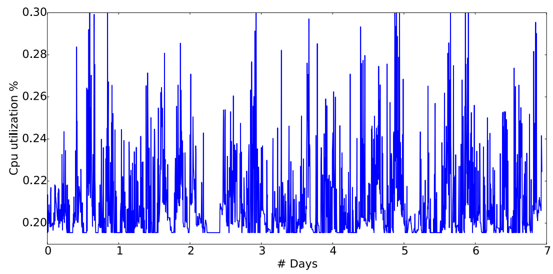

FIGURE 7. Example CPU consumption pattern of very extreme workloads.

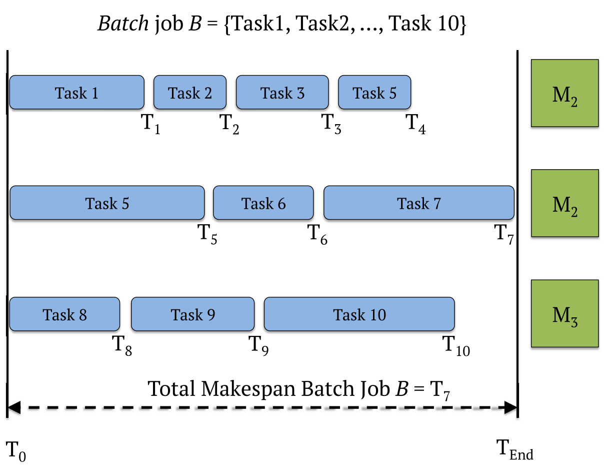

FIGURE 8. Makespan measuring workflow.

especially Batch jobs, since their determined end is a direct reflect of the achieved performance. Therefore, the performance is represented by the following key performance indicators (KPIs):

- **Queue times**: How long jobs J_j, composed of n_j tasks, are waiting in the queue until their first task $q_{(1)j}$ and their last task $q_{(n_j)j}$ are deployed.
- **Makespan**: How long does it take from the submission of the job until its completion, as shown in Figure 8. Thus, the makespan of the job J_j may be described as follows: $\mathcal{C}_j = q_{(n_j)j} + d_j$.

D. ENERGY MODEL

In this work, the energy consumption is measured as follows:

$$EC(\mathcal{T}) = \sum_{t\in\Delta} \sum_{m\in\mathcal{M}} \mathcal{P}(e_{m,t})\,\delta$$

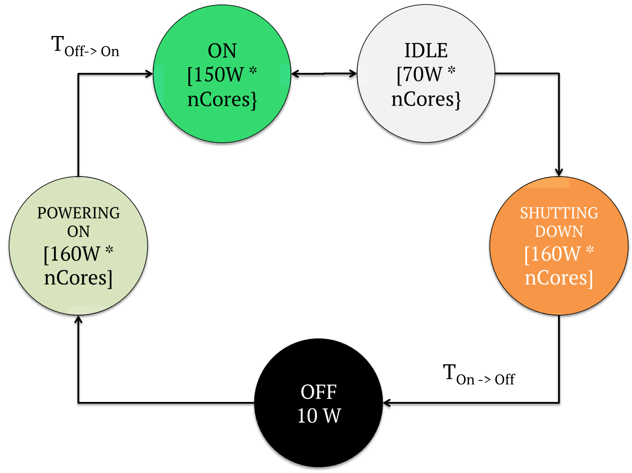

FIGURE 9. Machine power states.

Let $t \in \Delta = \{\delta, 2\delta, \ldots, \mathcal{T}\}$, where $\mathcal{T}$ is the total simulation time. For each period of time δ, the state of each machine $e_{m,t}$ is measured, and the energy consumption is computed depending of the power consumption $\mathcal{P}(e)$ of that particular state. The power states considered in this work $e \in$ {On, Idle, Switching}, where On denotes the power consumption of a machine when executing a task, Idle when the machine is waiting for an incoming job to be executed, and Switching if the machine is shutting-down or switching-on, as shown in Figure 9.

E. SIMULATION TOOL

To execute all simulations, a simulation tool able to simulate large-scale infrastructures, usually composed of thousands of nodes is required. Moreover, this tool must provide results in terms of performance and energy consumption. Several simulations tools have been analysed, including CloudSim [49], CloudSched [50], GreenCloud [50]. Some of them present serious limitations in terms of performance when large-scale infrastructures are considered, while others only focus on networking subsystems.

Authors presented and validated their own simulation tool, called SCORE [41], [45], that has been widely used in several works since then [41]–[44].

V. RESULTS

This section shows results for monolithic, two-level and shared-states resource managers. To this end we show tables where most relevant KPIs are presented for each scenario. Tables headings use the following abbreviations:

- Monolithic resource manager results: $\overline{q_1}$ represents the average job queue time until the first task is scheduled (the lower, the better the user experience). $\overline{q_n}$ denotes the average job queue time until the last task is scheduled (the lower, the better the user experience). $\overline{S_j}$ and $\overline{S_t}$ denote the average of job and tasks scheduling operations required by jobs to be fully scheduled, respectively

TABLE 1. Traditional data centres and monolithic resource manager: performance and energy-efficiency results of the batch workload.

Power-off policy	$\overline{q_1}$ (s)	$\overline{q_n}$ (s)	$\overline{S_j}$	$\overline{S_t}$	$\overline{\mathcal{C}}$ (s)	E_s (%)	SD	$\overline{E_{s,sd}}$ (kWh)	$t_{min,sd}$ (s)
Ordinary workload									
Keep on	0.23	0.23	1.00	413.70	39.98	0.00	0	0.00	0.00
Aggressive	1.20	8.83	5.90	713.80	52.81	59.15	7199	14.16	6.24
Random	0.95	6.15	4.60	624.90	48.88	58.21	5627	17.84	7.58
Margin	0.23	0.23	1.00	413.70	39.98	50.84	1692	50.26	6.80
Exp	0.23	0.23	1.00	414.10	39.99	54.55	1798	50.76	6.80
Gamma	0.23	0.23	1.00	414.10	39.99	54.45	1789	50.89	6.80
BF-size	0.59	3.15	1.60	499.60	44.45	57.62	3294	29.65	330.83
BF-time	0.56	3.00	1.90	505.10	44.41	57.67	3379	28.97	141.84
Extreme workload									
Keep on	1.29	1.29	1.00	416.40	39.98	0.00	0	0.00	0.00
Aggressive	4.55	24.17	7.80	1070.50	79.06	58.94	16618	6.03	6.54
Random	3.46	15.60	6.10	833.90	81.81	57.96	12536	7.82	6.19
Margin	1.29	1.30	1.00	416.90	39.99	50.74	1702	49.81	6.81
Exp	1.36	1.67	1.00	427.30	40.44	54.37	2089	43.75	5.89
Gamma	1.37	1.68	1.00	427.70	40.46	54.32	2086	43.76	6.49
BF-size	2.01	5.86	1.90	513.80	47.63	56.24	5237	18.35	39.84
BF-time	2.20	6.87	2.60	541.10	46.29	56.63	6264	15.45	24.75
Very extreme workload									
Keep on	4.76	4.76	1.00	415.30	40.12	0.00	0	0.00	0.00
Aggressive	16.26	72.87	9.10	1307.90	119.75	59.08	20093	4.94	4.91
Random	12.54	48.92	7.40	1052.40	89.09	57.94	14335	6.83	5.87
Margin	4.82	5.06	1.00	420.30	40.36	50.85	1705	49.83	8.22
Exp	5.57	8.44	1.00	460.20	42.91	54.48	2483	37.13	8.22
Gamma	5.51	8.11	1.00	455.60	42.67	54.36	2432	37.77	8.22
BF-size	7.39	18.61	3.70	591.50	55.91	55.34	6094	15.34	20.60
BF-time	8.84	27.08	4.70	705.70	65.12	56.31	8418	11.40	24.76

(the lower, the better to avoid scheduling bottlenecks). $\overline{\mathcal{C}}$ is the average job makespan (the lower, the better the user experience). $\boldsymbol{E_s}$ represents the percentage of energy saved (the higher, the better). ***SD*** is the total number of shut-down operations (the lower, the better, as it causes hardware stress). $\overline{\boldsymbol{E_{s,sd}}}$ denotes the energy saved per shut-down operation (the higher, the better), whilst $\boldsymbol{t_{min,sd}}$ represents the minimum time a machine has spent powered-off as cause of a shut-down operation (the higher, the better).

- Two-level resource manager results, the previous abbreviations plus: $\boldsymbol{R_{lock}}$ represents the percentage of resources that are blocked due to the pessimistic-blocking strategy of Two-level resource managers. $\boldsymbol{S_{ret}}$ denotes the number of scheduling operations retried due to lack of resources in the copy of the cluster state (the lower, the better to avoid scheduling bottlenecks). $\boldsymbol{J_{to}}$ represents the total number of timed-out jobs.
- Shared-state resource manager results, the previous abbreviations plus: $\boldsymbol{J_{conf}}$ $\boldsymbol{T_{conf}}$ represent the total number of jobs and tasks conflicts, respectively (the lower, the better for the overall performance).

A. MONOLITHIC RESOURCE MANAGER RESULTS

Tables 1 and 2 present the results both in terms of performance and energy consumption for ordinary and hyper-scale data centres, respectively, when typical, extreme and very extreme workloads are executed.

For ordinary data centres and workloads, in terms of performance, the Aggressive policy presents a serious impact in terms of queue times, forcing the jobs to spend approximately 6x (0.23s vs 1.20s) and 40x (0.23s vs 8.83s) more time in queue until their first and last tasks are scheduled, respectively. The cause of such delay is the need to retry the scheduling decisions due to the lack of available resources. The increment of approximately 500% (1 vs 5.9) in the number of scheduling operations performed for each job clearly confirms this behaviour. Notwithstanding, the number of scheduling operations of tasks increments only by 70% (413 vs 713), presenting a tail behaviour, which means that a relatively low number of tasks need to be rescheduled multiple times. All this poor scheduling performance leads to an increment of approximately 30% in terms of makespan (39.98s vs 52.81s). In contrary, fine-tuned strategies, such as Margin, Exponential (Exp) and Gamma, do not lead to performance penalisation.

Adaptive policies without fine-tuning, such as BullFighter, achieve relatively acceptable results in terms of performance, but the negative performance impact is not negligible, multiplying by ~3x (0.23s vs 0.56s) and ~13x (0.23s vs 3s) the time that jobs spend in queue until their first and last tasks are scheduled, respectively.

Still for ordinary data centres, in terms of energy efficiency, the trend is clear: the more aggressive the energy policy is, the lower the energy consumption. Nonetheless, fine-tuned policies, such as Gamma and Exponential, consume only 4.6% more energy than the Aggressive policy, while keeping the performance levels of the data centre. The quality of the shut-down operations is well reflected by the energy saved in each shut-down operation, which also reflects the

TABLE 2. Traditional data centres and monolithic resource manager: performance and energy-efficiency results of the batch workload.

Power-off policy	$\overline{q_1}$ (s)	$\overline{q_n}$ (s)	$\overline{S_j}$	$\overline{S_t}$	$\overline{C}$ (s)	E_s (%)	SD 10^3	$\overline{E_{s,sd}}$ (kWh)	$t_{min,sd}$ (s)
Ordinary workload									
Keep on	0.23	0.23	1	414	39.97	0.00	0	0.00	0.00
Aggressive	125.80	1789.45	25	2709	1639.66	25.69	2.54	21.93	5.01
Random	132.23	1873.22	26	2746	1845.28	25.53	1.99	27.87	3.19
Margin	0.23	0.23	1	414	39.97	5.68	1.26	9.67	9.28
Exponential	0.23	0.23	1	414	39.97	13.88	1.53	19.50	8.60
Gamma	0.23	0.23	1	414	39.97	13.80	1.52	19.43	9.28
BF-size	120.31	1733.55	24	2463	1885.78	25.09	1.79	31.19	533.46
BF-time	120.80	1727.36	24	2546	1627.23	25.35	1.88	29.80	426.69
Extreme workload									
Keep on	1.28	1.28	1	413	39.91	0.00	0	0.00	0.00
Aggressive	136.17	1923.84	24	2755	1744.86	25.78	2.51	22.17	5.25
Random	123.78	1708.53	23	2627	1514.49	25.54	1.94	28.36	5.66
Margin	1.28	1.28	1	413	39.91	5.68	1.26	9.67	8.47
Exponential	1.28	1.28	1	413	39.91	13.99	1.53	19.62	8.47
Gamma	1.28	1.28	1	413	39.91	13.96	1.53	19.58	8.47
BF-size	151.26	2281.32	24	2640	2012.71	25.31	1.47	37.61	498.41
BF-time	141.09	2062.20	24	2672	1815.58	25.38	1.67	34.39	1131.50
Very extreme workload									
Keep on	4.76	4.76	1	415	40.02	0.00	0	0.00	0.00
Aggressive	171.14	2193.08	24	2772	2354.66	25.79	2.62	21.27	4.75
Random	177.50	2253.10	23	2702	2251.54	25.60	2.09	26.79	4.86
Margin	4.76	4.76	1	415	40.02	5.68	1.26	9.64	9.57
Exponential	4.77	4.77	1	415	40.02	14.47	1.57	19.79	9.57
Gamma	4.76	4.77	1	415	40.02	14.10	1.54	19.58	9.57
BF-size	173.41	2201.06	22	2589	2001.92	25.08	1.78	31.12	2814.01
BF-time	165.58	2084.83	23	2666	2181.53	25.26	1.92	29.24	969.96

stress imposed on the hardware. The best results in terms of this metric are achieved by the fine-tuned policies, which perform less than 25% of shut-down operations compared to those performed by the Aggressive policy (~7,200 vs ~1800), followed by the adaptive policies, whose number of shut-down operations is less than half that of the Aggressive policy (~7,200 vs ~3,300). It must be noticed that adaptive policies present a good foreseeing behaviour, represented by the minimum time a machine spend shut-down for a single shut-down cycle (330s vs 6.80s of fine-tuned policies) which invites us to develop new policies of this family.

Regarding extreme scenarios, the less predictable workload peaks intensify the negative performance impact of the energy-efficiency policies. This trend can be stated in the number of re-scheduling operations needed to fully schedule a job. When the Aggressive policy is applied, jobs need, in average, 7.8 and 9.1 scheduling operations in extreme and very extreme workloads, compared to the 5.9 scheduling operations of ordinary workloads.

Fine-tuned policies, such as Margin, Exponential and Gamma cause minor performance deterioration in extreme workloads. The Margin policy presents the most conservative behaviour, causing virtually no performance degradation, while consuming ~3.5% more energy than Exponential and Gamma. Exponential and Gamma achieves to reduce such amount of energy consumption by incrementing the time jobs spend in queue until their last task is deployed in approximately 20% (1.29s vs 1.67s), while having a minor impact (less than 10%) in the time jobs spend in queue until their first task is deployed (1.29s vs 1.36s). The less stable pattern of extreme workloads leads to a higher number of shut-down operations: the Aggressive policy performs more than double of shut-down operations than for ordinary workloads (~16,600 vs ~7,200), whilst the adaptive policies, represented by the Bullfighter policy, perform slightly less than double (~3,300 vs 6,200). The Margin policy provides the same numbers than for ordinary workloads, whilst Exponential and Gamma perform approximately 15% more shut-down operations (1,800 vs. 2,100). The quality of shut-down operations, represented by the kW saved per shut-down operation, follows the same trend. Very extreme workloads increment the presented trends of performance penalisation.

For hyper-scale data centres, the penalisation in terms of performance is much more extreme, which makes adaptive energy-efficiency behave like the Aggressive policy, and ineffective for real application.

On the one hand, both the Aggressive and adaptive policies achieve the highest energy savings (~25%) at the cost of unacceptable job queue times of more than two minutes until the first task is scheduled, and more than 30 minutes until the last task is scheduled, compared to the 0.23 seconds of the current trend of not application of any efficiency policy.

On the other hand, the Margin policy becomes too conservative for these data centres, achieving a low level of energy reduction (5.68%), whilst Exponential and Gamma policies neither cause negative performance impact, and consume approximately 10% less energy.

From these results we can conclude that, for data centres which employ Monolithic resource managers, only fine-tuned energy-efficiency policies based on statistical

TABLE 3. Traditional data centres and two-level resource manager: performance and energy-efficiency results of the batch workload.

Power-off policy	$\overline{q_1}$ (s)	$\overline{q_n}$ (s)	R_{lock} (%)	S_{ret}	J_{to}	$\overline{C}$ (s)	E_s (%)	SD 10^3	$\overline{E_{s,sd}}$ (kWh)	$t_{min,sd}$ (s)
Ordinary workload										
Keep on	0.26	0.26	6.71	19	0	41.99	0.00	0	0.00	0.00
Aggressive	0.96	3.15	0.61	4471	0	51.46	58.39	18.78	5.25	8.31
Random	0.73	2.08	0.64	3248	0	46.88	57.48	13.66	7.10	8.76
Margin	0.26	0.26	1.22	8	0	42.01	50.12	1.85	46.34	532.20
Exponential	0.27	0.29	0.85	58	0	42.09	53.70	2.23	40.86	147.94
Gamma	0.27	0.29	0.85	59	0	42.10	53.75	2.29	39.93	64.20
BF-size	0.41	0.93	0.83	1231	0	43.67	55.05	5.32	17.74	73.75
BF-time	0.41	0.89	0.77	1268	0	43.82	55.65	5.78	16.31	37.34
Extreme workload										
Keep on	1.18	1.21	6.43	97	0	41.76	0.00	0	0.00	0.00
Aggressive	3.06	9.07	0.66	7427	0	55.46	58.43	28.08	3.55	8.84
Random	2.21	5.00	0.65	4498	0	49.35	57.23	19.42	5.01	8.44
Margin	1.16	1.17	1.15	44	0	41.84	50.08	2.84	30.01	18.45
Exponential	1.18	1.24	0.80	171	0	42.05	53.71	3.68	24.71	10.67
Gamma	1.18	1.23	0.83	148	0	42.01	53.57	3.51	25.84	18.09
BF-size	1.46	2.47	0.88	1447	0	44.22	54.11	6.97	13.26	37.43
BF-time	1.49	2.52	0.84	1613	0	44.49	54.74	8.42	11.07	34.95
Very extreme workload										
Keep on	3.68	3.96	6.34	566	0	43.12	0.00	0	0.00	0.00
Aggressive	78.47	1760.83	10.67	193848	718	58990	49.10	40.37	2.12	10.72
Random	5.93	15.61	0.88	9769	0	61.31	56.11	29.83	3.20	10.10
Margin	3.53	3.77	1.13	592	0	43.86	49.86	6.83	12.39	10.00
Exponential	3.71	4.77	0.85	1960	0	46.05	53.32	10.61	8.65	9.77
Gamma	3.65	4.29	0.88	1445	0	45.07	52.98	9.47	9.55	9.77
BF-size	3.94	6.36	1.42	2231	0	47.37	48.31	9.91	8.28	20.69
BF-time	4.20	7.23	1.21	3011	0	48.68	50.88	12.85	6.71	14.98

distributions can reduce the energy consumption without a negative impact in performance in hyper-scale data centres. This is due to these data centres utilising more intensively the resources. In such scenarios, workload peaks induce a high number of power-on operations, since not enough resources are available. In traditional data centres, which employ less intensively the resources, even simple adaptive policies, which try to foresee the incoming workload without fine tuning, provide interesting results.

B. TWO-LEVEL RESOURCE MANAGER RESULTS

In this section, we evaluate the result of the application of the energy-efficiency policies in data centres which equip a Two-level resource manager, which blocks the cluster when one of the parallel scheduling agents is making scheduling decisions. 4 scheduling agents are working in parallel to service Batch jobs, and 1 scheduling agent is responsible for the scheduling of Service jobs.

The first difference with the Monolithic resource managers becomes evident for traditional data centres, as shown in Table 3: Two-level resource managers tend to decrease the difference in terms of the time jobs spend in queue between very aggressive and conservative energy-efficiency policies in low and medium-stressed scenarios, at the cost of approximately two percent more energy consumed.

The application of conservative energy-efficiency policies in traditional data centres, such as Margin, Exponential or Gamma, leads to longer queue times for jobs of ordinary workloads, which wait in queue approximately 10% longer until their first task is scheduled, compared to Monolithic schedulers (0.23s vs. 0.27s). This impact raise to 20% for the queue time until all their tasks are scheduled (0.23s vs 0.29s). These longer queue times have a direct impact on the makespan (~40s vs. 42s).

On the other hand, the negative performance impact of the Aggressive policy decreases. The time jobs spend in queue until their last task is scheduled is approximately 3 times lower (8.83s vs 3.15s), which leads to the reduction of 2% of the makespan (52.81s vs 51.46s). This trend fades for extreme and even reverts for very extreme workloads, where the Aggressive policy is unable to service all the incoming workload, as can be seen in the makespan results (almost 17 hours compared to the ~2 minutes of the Monolithic resource manager). This comes to the cost of less than 1% more energy consumed for ordinary workloads, and ~10% more energy consumed in very extreme scenarios.

In terms of energy, the pessimistic blocking strategy implemented by Two-level resource managers leads to a slightly higher energy consumption (+2%) and a higher number of shut-down operations, linked to the number of scheduling operations that were not able to find available resources. This means that, the more stressful the scenario (less available resources or more extreme arrival patterns), the higher the number of shut-down operations, and, in consequence, the lesser the quality of each shut-down operation. A good example of this trend can be found for Hyper-scale data centres when the Random energy-efficiency policy is applied: while the number of shut-down operations is approximately 30% lesser than that of the Aggressive energy-efficiency policy for ordinary workloads, both number almost equal for very extreme workloads.

TABLE 4. Hyper-scale data centres and Two-level resource manager: Performance and energy-efficiency results of the batch workload.

Power-off policy	$\overline{q_1}$ (s)	$\overline{q_n}$ (s)	R_{lock} (%)	S_{ret}	J_{to}	$\overline{\mathcal{C}}$ (s)	E_s (%)	SD 10^3	$\overline{E_{s,sd}}$ (kWh)	$t_{min,sd}$ (s)
Ordinary workload										
Keep on	0.26	0.26	3.98	19	0	41.98	0.00	0	0.00	0.00
Aggressive	1.58	6.44	1.16	5984	0	52.97	24.86	16.23	3.32	6.59
Random	1.19	4.75	1.07	4211	0	49.84	24.41	11.60	4.56	6.59
Margin	0.26	0.26	3.24	7	0	42.00	5.47	1.49	8.05	6.59
Exponential	0.26	0.26	2.14	8	0	42.00	13.44	1.76	16.67	6.59
Gamma	0.26	0.26	2.14	6	0	42.00	13.49	1.75	16.78	6.59
BF-size	0.57	2.07	1.13	1660	0	44.94	22.53	4.62	10.63	48.30
BF-time	0.62	2.14	1.13	1808	0	45.23	22.68	5.33	9.32	6.59
Extreme workload										
Keep on	1.21	1.23	4.09	95	0	42.07	0.00	0	0.00	0.00
Aggressive	234.68	$4.85 * 10^3$	4.54	14.12	8	$34.98 * 10^3$	23.04	23.16	2.19	10.60
Random	136.28	$1.57 * 10^3$	2.98	7.35	1	$20.38 * 10^3$	23.23	16.13	3.16	9.60
Margin	1.20	1.20	3.32	33	0	42.14	5.38	2.08	6.07	56.21
Exponential	1.20	1.20	2.21	32	0	42.14	13.30	2.35	12.98	46.73
Gamma	1.20	1.20	2.22	33	0	42.14	13.26	2.35	13.06	40.32
BF-size	1.60	3.61	1.28	1769	0	45.58	21.33	5.71	8.16	41.91
BF-time	1.73	4.17	1.26	2526	0	46.93	22.15	7.15	6.79	22.64
Very extreme workload										
Keep on	3.64	3.91	3.72	536	0	43.17	0.00	0	0.00	0.00
Aggressive	$2.97 * 10^3$	$24.34 * 10^3$	23.12	422613	1900	$243.53 * 10^3$	12.64	21.26	1.38	13.78
Random	$1.63 * 10^3$	$19.08 * 10^3$	13.98	227344	1092	$140.65 * 10^3$	17.16	20.94	1.89	11.83
Margin	3.47	3.56	3.04	219	0	43.37	5.35	5.08	2.35	13.79
Exponential	3.48	3.56	1.99	229	0	43.38	13.26	5.44	5.37	10.34
Gamma	3.47	3.56	2.05	224	0	43.38	13.06	5.38	5.35	11.84
BF-size	4.06	6.81	1.63	2481	0	48.19	18.31	9.01	4.46	17.89
BF-time	542.31	$6.01 * 10^3$	6.10	89544	544	$53.64 * 10^3$	17.66	11.17	3.33	492.24

The higher number of scheduling agents of the Two-Level resource manager achieves a slightly better behaviour than that of the Monolithic schedulers for stressed scenarios.

In traditional data centres, whose resource utilisation is inferior than that of hyper-scale data centres, this resource manager could mitigate the negative effects of extreme workloads, especially when aggressive policies are applied. It also reduces the negative effects of very extreme workloads under conservative energy policies.

In hyper-scale data centres, the resource-offering and pessimistic-blocking strategy of these resource managers make these improvements minor, or even disadvantages when aggressive policies are combined with very extreme workloads, as shown in Table 4.

These results show a clear trend: whilst Two-level resource managers can improve the data-centre performance for more extreme workload arrival patterns and utilisation rates than Monolithic schedulers, Two-level resource managers are unable to handle successfully very extreme scenarios: very extreme arrival patterns combined with high resource-utilisation rates and aggressive energy-efficiency policies.

C. SHARED-STATE RESOURCE MANAGER RESULTS

Finally, we analyse the behaviour of efficiently data centres which equip a Shared-state resource manager, which tries to commit the decisions made by parallel scheduling agents to the central cluster without pessimistic blocking. If a conflict is found, the transaction is retried. For comparison purposes, the same configuration of 4 scheduling agents working in parallel to service Batch jobs, and 1 scheduling agent Service jobs is applied.

The results in terms of performance and energy efficiency are presented in Tables 5 and 6 for traditional and hyper-scale data centres, respectively. The number of timed-out jobs is not presented as it is 0.

In traditional data centres, the utilisation of a Shared-state model improves the performance results, especially when aggressive energy-efficiency policies are applied (see queue time results for Ordinary workloads and the Aggressive policy in Tables 1 and 5, ∼34ms vs ∼1.2s), while the energy reduction rates are maintained. This improvement becomes more acute when very extreme workloads are under consideration (see queue time results for Very extreme workloads and the Aggressive policy in Tables 1 and 5, ∼543ms vs ∼16s). Notwithstanding, even though the application of a Shared-state resource manager reduces queue times significantly, the makespan time results are not notably reduced when compared to those achieved by Monolithic schedulers for traditional data centres (see makespan time results for Extreme workloads and the Exp policy in Tables 1 and 5, 40.40s vs 41.78s). The application of the Shared-state model also leads to a higher number of shut-down operations, which directly affect the kWh saved per shut-down operation. Both are the consequences of the higher number of scheduling decisions made and the related conflicts, which may stress the hardware, especially when aggressive policies are applied (see #shut-down operations results for Ordinary workloads and the Aggressive policy in Tables 1 and 5, ∼16k vs ∼7k shut-down operations, respectively).

TABLE 5. Traditional data centres and shared-state resource manager: performance and energy-efficiency results of the batch workload.

Power-off policy	$\overline{q_1}$ (ms)	$\overline{q_n}$ (ms)	J_{conf}	T_{conf} 10^3	$\overline{\mathcal{C}}$ (s)	E_s (%)	SD 10^3	$\overline{E_{s,sd}}$ (kWh)	$t_{min,sd}$ (s)
Ordinary workload									
Keep on	0.51	0.96	375	10.18	40.14	0.00	0	0.00	0.00
Aggressive	34.11	317.55	1193	54.81	51.74	58.85	16.09	6.18	8.25
Random	24.6	219.64	967	42.24	49.87	57.95	11.89	8.19	7.64
Margin	0.53	1.65	383	10.26	40.53	50.69	1.69	50.17	7.92
Exponential	1.11	4.7	397	11.00	40.61	54.38	1.88	48.33	7.92
Gamma	1.14	5.02	396	11.47	40.62	54.31	1.89	48.17	7.50
BF-size	7.83	66.91	544	20.51	41.97	55.93	4.49	20.90	91.92
BF-time	8.92	78.49	590	22.39	42.27	56.44	5.09	18.82	64.56
Extreme workload									
Keep on	27.69	46.67	1128	36.37	40.40	0.00	0	0.00	0.00
Aggressive	111.33	811.1	2416	123.42	50.26	58.77	23.24	4.27	8.35
Random	85.63	549.02	2032	97.75	47.62	57.75	16.86	5.81	7.39
Margin	27.4	42.17	1121	36.37	41.52	50.74	1.70	50.03	8.35
Exponential	30.26	62.65	1175	39.61	41.78	54.27	2.35	38.92	8.35
Gamma	30.24	61.98	1176	40.15	41.77	54.19	2.33	39.24	8.35
BF-size	42.7	174.64	1396	55.06	43.39	55.49	5.84	16.18	23.82
BF-time	45.05	206.09	1464	59.85	43.86	56.13	6.98	13.57	21.66
Very extreme workload									
Keep on	275.11	445.73	2106	1151.43	40.49	0.00	0	0.00	0.00
Aggressive	543.34	2740.41	4131	1151.34	57.92	58.64	32.75	3.02	5.75
Random	451.62	1853.38	3513	1151.36	53.72	57.49	23.25	4.14	6.76
Margin	272.17	364.23	2083	1151.43	42.64	50.78	1.94	43.99	7.54
Exponential	299.15	549.54	2286	1151.41	43.63	54.16	4.20	22.54	7.54
Gamma	293.14	505.92	2236	1151.42	43.44	54.02	3.81	24.79	7.54
BF-size	318.75	730.73	2450	1151.43	45.00	53.87	7.38	12.39	33.07
BF-time	332.7	865.84	2599	1151.36	45.87	54.82	9.73	9.45	21.63

TABLE 6. Hyperscale data centres and shared-state resource manager: performance and energy-efficiency results of the batch workload.

Power-off policy	$\overline{q_1}$ (ms)	$\overline{q_n}$ (ms)	J_{conf}	T_{conf} 10^3	$\overline{\mathcal{C}}$ (s)	E_s (%)	SD 10^3	$\overline{E_{s,sd}}$ (kWh)	$t_{min,sd}$ (s)
Ordinary workload									
Keep on	0.39	0.85	381	10	40.09	0.00	0	0.00	0.00
Aggressive	$13.69 * 10^3$	$381.67 * 10^3$	14385	150	498.95	25.59	4.61	12.12	6.33
Random	$13.35 * 10^3$	$374.08 * 10^3$	13681	143	486.02	25.31	3.92	14.21	7.30
Margin	0.49	1.55	380	10	40.47	5.68	1.27	9.60	10.00
Exponential	0.50	1.61	383	10	40.48	13.88	1.54	19.35	10.00
Gamma	0.48	1.58	382	10	40.48	13.91	1.54	19.40	10.00
BF-size	$9.54 * 10^3$	$259.91 * 10^3$	10088	101	352.35	24.28	2.78	19.20	206.67
BF-time	$9.31 * 10^3$	$250.92 * 10^3$	10689	111	346.77	24.47	3.21	16.72	243.32
Extreme workload									
Keep on	28.67	47.99	1151	38	40.39	0.00	0	0.00	0.00
Aggressive	$13.31 * 10^3$	$354.90 * 10^3$	15863	189	486.76	25.73	4.49	12.66	3.12
Random	$11.69 * 10^3$	$314.88 * 10^3$	14155	175	445.17	25.40	3.70	15.46	4.26
Margin	28.43	41.59	1133	37	41.51	5.68	1.27	9.64	8.28
Exponential	28.43	41.74	1126	37	41.51	13.99	1.54	19.51	8.28
Gamma	28.48	41.63	1128	37	41.51	13.98	1.54	19.52	8.28
BF-size	$8.39 * 10^3$	$226.42 * 10^3$	10696	137	330.71	24.44	2.69	20.68	804.17
BF-time	$9.70 * 10^3$	$259.77 * 10^3$	12141	150	372.36	24.66	3.13	18.20	734.20
Very extreme workload									
Keep on	284.63	465.29	2144	96	40.71	0.00	0	0.00	0.00
Aggressive	$16.09 * 10^3$	$388.01 * 10^3$	16533	240	509.22	25.65	4.78	11.73	6.35
Random	$15.25 * 10^3$	$362.23 * 10^3$	15651	237	478.74	25.40	4.02	13.79	5.98
Margin	275.94	348.31	2086	93	42.74	5.69	1.28	9.50	6.86
Exponential	276.05	349.06	2088	94	42.74	14.35	1.57	19.56	6.86
Gamma	276.07	349.27	2081	94	42.74	14.09	1.56	19.36	6.86
BF-size	$9.89 * 10^3$	$223.64 * 10^3$	11489	203	311.62	24.18	3.14	17.13	629.92
BF-time	$13.54 * 10^3$	$337.12 * 10^3$	13501	217	434.87	24.63	3.73	14.50	41.35

It is also worth mentioning that the Shared-state not only improves all key performance indicators, but also is the only model that provides a predictable behaviour even for hyper-scale data centres, as reflected in the number of timed out jobs (see the Aggressive policy in Tables 6 and 4: 1900 vs 0, respectively). When aggressive policies are applied to these data centres, Shared-state resource managers reduce by ~1/3 the makespan (see makespan results for Very

TABLE 7. Environmental results of the application of the proposed energy-efficiency policies per year in the United States. For this table, the following parameters have been employed: a) 700 grams of CO_2 are emitted per kWh; b) the price kWh taken is 0.12$; and c) an average car is considered to emit 2.4 tons of CO_2 per year.

Data-Centre Type	Hyper-scale	Traditional	Total
Current Total Energy Consumption (B kWh)	18	52	**70**
PUE	1.1	2.0	1.77
IT Equipment Energy Consumption (B kWh)	16.36	26.00	42.36
Infrastructure Energy Consumption (B kWh)	1.64	26.00	27.64
Green Data centres Energy Consumption (B kWh)	15.72	37.84	**53.57**
IT Equipment Energy Saved (B kWh)	2.28	14.16	**16.43**
CO2 Saved (M Tons)	1.59	9.91	**11.50**
Costs Saved (B $)	0.27	1.70	**1.97**
M Cars Removed	0.66	4.1	**4.79**

extreme workloads and the BF-size policy in Tables 2 and 6, ~2001s vs ~311s, respectively), and by approximately 1 and 2 orders of magnitude the queue times when compared to the Monolithic and Two-level approaches, respectively.

However, it must be highlighted that the application of aggressive energy- efficiency policies in hyper-scale data centres may skyrocket queue times even for Shared-state resource managers (see makespan results in Table 6 for Margin and Aggressive policies: ~43s vs ~509s, respectively). The very same negative impact is applied when the forecast is not accurate enough (see makespan results in Table 6 for the BF-time and Gamma policies: ~435s vs ~43s, respectively).

D. ENVIRONMENTAL RESULTS FOR USA DATA CENTRES

In order to illustrate the magnitude of the savings that could be achieved by applying switch-off energy policies such as the ones presented for hyper-scale and traditional data centres, we present a simple extrapolation of these results to USA data centres, based on the data presented by [3].

To compute energy savings, we only consider the energy savings of IT Equipment (infrastructure energy consumption such as cooling could also be saved, but not included in these calculations). Current consumption of IT Equipment of data centres is computed from current consumption and the Power Usage Effectiveness (PUE).

According to results presented in previous sections, we can conclude that ordinary data centres could save ~55% and hyper-scale data centres could save ~14% of energy consumed by IT equipment. Therefore, we computed these energy savings, and consequently the consumption of data centres applying these policies.

To this end, we have considered that 700 grams of CO_2 are emitted per kWh generated considering all generation sources (including greener generation and others as coal-based generation), a cost of 0.12 USD per kWh consumed, and the CO_2 emissions of an average car. An average car is considered to emit 0.124 grams of CO_2 and to travel 20,000 kms per year.

So, taking into account that current energy consumption of USA data centres sums up to 70 billion of kWh ($70 * 10^9$) [3], if energy policies were applied, energy consumption should reduce to $53.57 * 10^9$ kWh, which would mean that 11.5 million of tons of CO_2 would not be emitted to the atmosphere, or an equivalent of removing 4.79 million of cars, and saving 1.97 billion of USD only in energy bills.

It is important to notice, that power consumption changes proposed by applying these policies could provoke changes over power grid, so power suppliers should be aware of the possibility of sudden consumption changes. However the most aggressive changes that these policies could provoke over the whole USA power Grid would stand around 1.2 MWh. According to the U. S. Energy Information Administration [51], these changes could be easily absorbed by the power grid, as 1.2MWh would be around 0.0003% of the minimum US power grid generation.

VI. CONCLUSION

Various methodologies for tackling energy saving for data centres in hyper-scale or ordinary environments have been presented.

This way, authors have empirically proven that the application of such energy-policies at a software level could reduce a very important amount of the energy consumed by these greedy infrastructures, and consequently reduce the CO_2 produced and emitted to the atmosphere. Depending on the scenario, energy savings can vary from ~15% to as much as ~60%.

Moreover, we have shown that all scenarios are suitable of saving energy by applying energy policies, but certain combinations of the data-centre resource manager plus the energy policy are better for scenarios where the data-centre load follows one pattern or another, and depending on the data centre scale as well. On the one hand, for hyper-scale data centres that deal with extreme loads, shared-state resource managers combined with energy saving policies could keep a high performance while saving energy. On the other hand, if a traditional data centre deals with an extreme load, two-level resource managers could also be a good approach, keeping good performance and saving important amounts of energy.

Finally, it is important to notice that for very extreme scenarios, performance could be deteriorated if some energy policies are selected, so data-centres' managers should make decisions based on previous simulations. This is especially noticeable for the bigger and the more extreme workloads. In such cases, the negative impact over the performance is only avoidable if energy policies are very fine-tuned and predictions are designed for specific load patterns.

However, it must be taken into account that large Internet companies depend on the extreme performance of their data centres. Therefore, even a minimum performance impact could impose a cost opportunity that could be used by competitors to gain market share. Thus, the energy-efficiency policy to be used, if used at all, depends on the requirements of the data-centre operator, needing to balance the trade-off between the economic and environmental benefits related to reducing energy consumption and the market opportunity. As future work, we are working on the development of smarter energy-efficiency models that would decrease significantly this performance impact, and to include the impact of the renewable energy-production models to the data-centre environment.

As a summary, the novelties presented in this paper are:

- the analysis of the application of several energy policies, from simple to fine-tuned and adaptive, to various data centres, from ordinary to hyper-scale, and switching from ordinary to extreme and very extreme workloads and top three resource managers, and
- the computation of the energy savings that these policies could imply if applied to USA data centres, and consequently the positive impact on the environment.

REFERENCES

[1] *Energy, Transport and Environment Indicators*. Office Off. Publications Eur. Communities, Commission Européenne, Eurostat Union Européenne, Brussels, Belgium, 2017.

[2] M. Allen, M. Babiker, Y. Chen, H. de Coninck, S. Connors, R. van Diemen, O. P. Dube, K. L. Ebi, F. Engelbrecht, M. Ferrat, J. Ford, P. Forster, S. Fuss, T. G. Bolaños, J. Harold, O. Hoegh-Guldberg, J.-C. Hourcade, D. Huppmann, and K. Zickfeld, ''Summary for policymakers,'' in *Global Warming of 1.5 °C. An IPCC Special Report on the Impacts of Global Warming of 1.5 °C Above Pre-Industrial Levels and Related Global Greenhouse Gas Emission Pathways, in the Context of Strengthening the Global Response to the Threat of Climate Change, Sustainable Development, and Efforts to Eradicate Poverty*, V. Masson-Delmotte, P. Zhai, H.-O. Pörtner, D. Roberts, J. Skea, P. R. Shukla, A. Pirani, W. Moufouma-Okia, C. Péan, R. Pidcock, S. Connors, J. B. R. Matthews, Y. Chen, X. Zhou, M. I. Gomis, E. Lonnoy, T. Maycock, M. Tignor, and T. Waterfield, Eds. Geneva, Switzerland: World Meteorological Organization, 2018, p. 32.

[3] A. Shehabi, S. J. Smith, D. A. Sartor, R. E. Brown, M. Herrlin, J. G. Koomey, E. R. Masanet, N. Horner, I. L. Azevedo, and W. Lintner, ''United states data center energy usage report,'' Lawrence Berkeley Nat. Lab., Berkeley, Ca, USA, Tech. Rep. LBNL-1005775, Jun. 2016.

[4] P. D. Saundry, ''Review of the united states energy system in transition,'' *Energy, Sustainability Soc.*, vol. 9, no. 1, p. 4, Jan. 2019, doi: 10.1186/s13705-018-0178-8.

[5] R. K. Sharma, C. E. Bash, C. D. Patel, R. J. Friedrich, and J. S. Chase, ''Balance of power: Dynamic thermal management for Internet data centers,'' *IEEE Internet Comput.*, vol. 9, no. 1, pp. 42–49, Jan. 2005.

[6] N. El-Sayed, I. A. Stefanovici, G. Amvrosiadis, A. A. Hwang, and B. Schroeder, ''Temperature management in data centers: Why some (might) like it hot,'' *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 1, p. 163, Jun. 2012.

[7] X. Fan, W.-D. Weber, and L. A. Barroso, ''Power provisioning for a warehouse-sized computer,'' *ACM SIGARCH Comput. Archit. News*, vol. 35, no. 2, p. 13, Jun. 2007.

[8] A. Miyoshi, C. Lefurgy, E. Van Hensbergen, R. Rajamony, and R. Rajkumar, ''Critical power slope: Understanding the runtime effects of frequency scaling,'' in *Proc. 16th Int. Conf. Supercomput. (ICS)*, 2002, pp. 35–44.

[9] D. G. Andersen and S. Swanson, ''Rethinking flash in the data center,'' *IEEE Micro*, vol. 30, no. 4, pp. 52–54, Jul. 2010.

[10] M. E. Femal and V. W. Freeh, ''Boosting data center performance through non-uniform power allocation,'' in *Proc. 2nd Int. Conf. Autonomic Comput. (ICAC)*, 2005, pp. 250–261.

[11] M. Isard, V. Prabhakaran, J. Currey, U. Wieder, K. Talwar, and A. Goldberg, ''Quincy: Fair scheduling for distributed computing clusters,'' in *Proc. ACM SIGOPS 22nd Symp. Oper. Syst. Princ. (SOSP)*, 2009, pp. 261–276.

[12] C. Delimitrou, D. Sanchez, and C. Kozyrakis, ''Tarcil: Reconciling scheduling speed and quality in large shared clusters,'' in *Proc. 6th ACM Symp. Cloud Comput. (SoCC)*, 2015, pp. 97–110.

[13] C. Delimitrou and C. Kozyrakis, ''Paragon: QoS-aware scheduling for heterogeneous datacenters,'' *ACM SIGPLAN Notices*, vol. 48, no. 4, p. 77, Apr. 2013.

[14] M. Zaharia, D. Borthakur, J. Sen Sarma, K. Elmeleegy, S. Shenker, and I. Stoica, ''Delay scheduling: A simple technique for achieving locality and fairness in cluster scheduling,'' in *Proc. 5th Eur. Conf. Comput. Syst.*, 2010, pp. 265–278.

[15] A. Verma, L. Pedrosa, M. Korupolu, D. Oppenheimer, E. Tune, and J. Wilkes, ''Large-scale cluster management at Google with borg,'' in *Proc. 10th Eur. Conf. Comput. Syst. (EuroSys)*, 2015, p. 18.

[16] C. Delimitrou and C. Kozyrakis, ''Quasar: Resource-efficient and QoS-aware cluster management,'' in *Proc. 19th Int. Conf. Architectural Support Program. Lang. Oper. Syst. (ASPLOS)*, 2014, pp. 127–144.

[17] X. Zhang, E. Tune, R. Hagmann, R. Jnagal, V. Gokhale, and J. Wilkes, ''CPI2: CPU performance isolation for shared compute clusters,'' in *Proc. 8th ACM Eur. Conf. Comput. Syst. (EuroSys)*, 2013, pp. 379–391.

[18] M. Schwarzkopf, A. Konwinski, M. Abd-El-Malek, and J. Wilkes, ''Omega: Flexible, scalable schedulers for large compute clusters,'' in *Proc. 8th ACM Eur. Conf. Comput. Syst. (EuroSys)*, 2013, pp. 351–364.

[19] B. Hindman, A. Konwinski, M. Zaharia, A. Ghodsi, A. D. Joseph, R. H. Katz, S. Shenker, and I. Stoica, ''Mesos: A platform for fine-grained resource sharing in the data center,'' in *Proc. NSDI*, vol. 11, 2011, p. 22.

[20] V. K. Vavilapalli, A. C. Murthy, C. Douglas, S. Agarwal, M. Konar, R. Evans, T. Graves, J. Lowe, H. Shah, and S. Seth, ''Apache Hadoop yarn: Yet another resource negotiator,'' in *Proc. 4th Annu. Symp. Cloud Comput.*, 2013, p. 5.

[21] Y. Chen, S. Alspaugh, and R. Katz, ''Interactive analytical processing in big data systems: A cross-industry study of mapreduce workloads,'' *Proc. VLDB Endowment*, vol. 5, no. 12, pp. 1802–1813, 2012.

[22] C. Reiss, J. Wilkes, and J. L. Hellerstein, ''Google cluster-usage traces: Format + schema,'' Google, Mountain View, CA, USA, version 2.1, Nov. 2011. [Online]. Available: https://github.com/google/cluster-data

[23] G. Ananthanarayanan, A. Ghodsi, A. Wang, D. Borthakur, S. Kandula, S. Shenker, and I. Stoica, ''Pacman: Coordinated memory caching for parallel jobs,'' in *Proc. 9th USENIX Conf. Netw. Syst. Design Implement.*, 2012, p. 20.

[24] C. Reiss, A. Tumanov, G. R. Ganger, R. H. Katz, and M. A. Kozuch, ''Heterogeneity and dynamicity of clouds at scale: Google trace analysis,'' in *Proc. 3rd ACM Symp. Cloud Comput.*, 2012, p. 7.

[25] K. Ren, Y. Kwon, M. Balazinska, and B. Howe, ''Hadoop's adolescence: An analysis of Hadoop usage in scientific workloads,'' *Proc. VLDB Endowment*, vol. 6, no. 10, pp. 853–864, Aug. 2013.

[26] A. Beloglazov and R. Buyya, ''Energy efficient resource management in virtualized cloud data centers,'' in *Proc. 10th IEEE/ACM Int. Conf. Cluster, Cloud Grid Comput.*, 2010, pp. 826–831.

[27] A. Beloglazov, J. Abawajy, and R. Buyya, ''Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing,'' *Future Gener. Comput. Syst.*, vol. 28, no. 5, pp. 755–768, May 2012.

[28] A. Beloglazov and R. Buyya, ''Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers,'' *Concurrency Comput., Pract. Exper.*, vol. 24, no. 13, pp. 1397–1420, 2012.

[29] S. Sohrabi, A. Tang, I. Moser, and A. Aleti, ''Adaptive virtual machine migration mechanism for energy efficiency,'' in *Proc. 5th Int. Workshop Green Sustain. Softw. (GREENS)*, 2016, pp. 8–14.

[30] T. H. Duong-Ba, T. Nguyen, B. Bose, and T. T. Tran, ''A dynamic virtual machine placement and migration scheme for data centers,'' *IEEE Trans. Services Comput.*, to be published.

[31] F.-H. Tseng, X. Wang, L.-D. Chou, H.-C. Chao, and V. C. M. Leung, ''Dynamic resource prediction and allocation for cloud data center using the multiobjective genetic algorithm,'' *IEEE Syst. J.*, vol. 12, no. 2, pp. 1688–1699, Jun. 2018.

[32] A. Jakóbik, D. Grzonka, and J. Kołodziej, ''Security supportive energy aware scheduling and scaling for cloud environments,'' in *Proc. Eur. Conf. Modelling Simulation (ECMS)*, Budapest, Hungary, May 2017, pp. 583–590.

[33] H. Amur, J. Cipar, V. Gupta, G. R. Ganger, M. A. Kozuch, and K. Schwan, ''Robust and flexible power-proportional storage,'' in *Proc. 1st ACM Symp. Cloud Comput. (SoCC)*, 2010, pp. 217–228.

[34] R. T. Kaushik and M. Bhandarkar, ''Greenhdfs: Towards an energy-conserving, storage-efficient, hybrid Hadoop compute cluster,'' in *Proc. USENIX Annu. Tech. Conf.*, 2010, p. 109.

[35] X. Luo, Y. Wang, Z. Zhang, and H. Wang, ''Superset: A non-uniform replica placement strategy towards high-performance and cost-effective distributed storage service,'' in *Proc. Int. Conf. Adv. Cloud Big Data*, Dec. 2013, pp. 139–146.

[36] T. V. T. Duy, Y. Sato, and Y. Inoguchi, "Performance evaluation of a green scheduling algorithm for energy savings in cloud computing," in *Proc. IEEE Int. Symp. Parallel Distrib. Process., Workshops Phd Forum (IPDPSW)*, Apr. 2010, pp. 1–8.

[37] Y. C. Lee and A. Y. Zomaya, "Energy efficient utilization of resources in cloud computing systems," *J. Supercomput.*, vol. 60, no. 2, pp. 268–280, Mar. 2010.

[38] F. Juarez, J. Ejarque, and R. M. Badia, "Dynamic energy-aware scheduling for parallel task-based application in cloud computing," *Future Gener. Comput. Syst.*, vol. 78, pp. 257–271, Jan. 2018.

[39] S. Ricciardi, D. Careglio, G. Santos-Boada, J. Sole-Pareta, U. Fiore, and F. Palmieri, "Saving energy in data center infrastructures," in *Proc. 1st Int. Conf. Data Compress., Commun. Process.*, Jun. 2011, pp. 265–270.

[40] A. Fernández-Montes, D. Fernández-Cerero, L. González-Abril, J. A. Álvarez-García, and J. A. Ortega, "Energy wasting at Internet data centers due to fear," *Pattern Recognit. Lett.*, vol. 67, pp. 59–65, Dec. 2015.

[41] D. Fernández-Cerero, A. Jakóbik, A. Fernández-Montes, and J. Kołodziej, "GAME-SCORE: Game-based energy-aware cloud scheduler and simulator for computational clouds," *Simul. Model. Pract. Theory*, vol. 93, pp. 3–20, May 2019.

[42] D. Fernández-Cerero, A. Fernández-Montes, and J. A. Ortega, "Energy policies for data-center monolithic schedulers," *Expert Syst. Appl.*, vol. 110, pp. 170–181, Nov. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417418303531

[43] D. Fernández-Cerero, A. Fernández-Montes, and F. Velasco, "Productive efficiency of energy-aware data centers," *Energies*, vol. 11, no. 8, p. 2053, Aug. 2018.

[44] D. Fernández-Cerero, A. Jakóbik, D. Grzonka, J. Kołodziej, and A. Fernández-Montes, "Security supportive energy-aware scheduling and energy policies for cloud environments," *J. Parallel Distrib. Comput.*, vol. 119, pp. 191–202, Sep. 2018.

[45] D. Fernández-Cerero, A. Fernández-Montes, A. Jakóbik, J. Kołodziej, and M. Toro, "SCORE: Simulator for cloud optimization of resources and energy consumption," *Simul. Model. Pract. Theory*, vol. 82, pp. 160–173, Mar. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1569190X18300030, doi: 10.1016/j.simpat.2018.01.004.

[46] A.-C. Orgerie, L. Lefèvre, and J.-P. Gelas, "Save watts in your grid: Green strategies for energy-aware framework in large scale distributed systems," in *Proc. 14th IEEE Int. Conf. Parallel Distrib. Syst.*, Dec. 2008, pp. 171–178.

[47] C. Reiss, A. Tumanov, G. R. Ganger, R. H. Katz, and M. A. Kozuch, "Heterogeneity and dynamicity of clouds at scale: Google trace analysis," in *Proc. ACM Symp. Cloud Comput. (SoCC)*, San Jose, CA, USA, Oct. 2012, pp. 1–13.

[48] Y. Cheng, A. Anwar, and X. Duan, "Analyzing Alibaba's co-located datacenter workloads," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 292–297.

[49] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Softw., Pract. Exper.*, vol. 41, no. 1, pp. 23–50, Aug. 2010.

[50] D. Kliazovich, P. Bouvry, and S. U. Khan, "GreenCloud: A packet-level simulator of energy-aware cloud computing data centers," *J. Supercomput.*, vol. 62, no. 3, pp. 1263–1283, Nov. 2010.

[51] U.S. Energy Information Administration. (2020). *U.S. Electricity Overview Demand, Forecast Demand, Net Generation, and Total Interchange*. [Online]. Available: https://www.eia.gov/beta/electricity/gridmonitor/expanded-view/electric_overview/US48/US48/ElectricityOverview-2

DAMIÁN FERNÁNDEZ-CERERO received the Ph.D. degree in computer science from the University of Seville, in 2018. He is currently a Lecturer and a Researcher with the University of Seville. He was granted with a Marie Curie Postdoctoral Fellowship with Dublin City University. He has collaborated in several National and European Projects and with European research groups in multiple research stages. As a result of his research interests, he has published numerous high-impact peer-review journal articles in various international conferences.

ALEJANDRO FERNÁNDEZ-MONTES received the B.E. degree in computer science and the M.Tech. and international Ph.D. degrees in software engineering from the University of Seville, Spain.

In 2006, he joined the Department of Computer Languages and Systems, University of Seville, as an Intern, and he became a Lecturer and an Assistant Professor, in 2013. In 2018, he became an Associate Professor (Professor Titular de Universidad). In 2008 and 2009, he was invited to work with the École Normale Supérieure de Lyon (ENS de Lyon) in saving energy solutions for Grid'5000 infrastructure. In 2012, he was invited to work at the Universitat Politécnica de Barcelona to share experiences in saving energy and doctoral methodologies. In 2016, he was invited to work with Shanghai Jiao Tong University, China, to initiate collaborations in distributed computing. He currently teaches and conducts research with the University of Seville, where he is also the Principal Investigator of several projects. His research interests include energy efficiency in distributed computing, applying prediction models to balance load, and applying on–off policies to the Internet data centers.

AGNIESZKA JAKÓBIK (KROK) received the M.Sc. degree in the field of stochastic processes from Jagiellonian University, Cracow, Poland, and the Ph.D. degree in the field of neural networks from the Cracow University of Technology, Poland, in 2003 and 2007, respectively. Since 2009, she has been an Assistant Professor with the Faculty of Physics, Mathematics, and Computer Science, Cracow University of Technology.