From Understanding to Greening the Energy Consumption of Distributed Systems

Anne-Cécile Orgerie

Habilitation à Diriger des Recherches

présentée devant l'École Normale Supérieure de Rennes

Spécialité : Informatique

au titre de l'École Doctorale MathSTIC (ED 601)

soutenue le 20-11-2020 par

Anne-Cécile Orgerie

From Understanding to Greening the Energy Consumption of Distributed Systems

Devant le jury composé de :

Mme Sara Bouchenak	Professeure à l'Institut National des Sciences Appliquées de Lyon, France	Examinatrice
M. Jesús Carretero	Professeur à l'Université Carlos III de Madrid, Espagne	Rapporteur
Mme Anne-Marie Kermarrec	Professeure à l'Ecole Polytechnique Fédérale de Lausanne, Suisse	Examinatrice
M. Martin Quinson	Professeur à l'Ecole Normale Supérieure de Rennes, France	Président
Mme Catherine Rosenberg	Professeure à l'Université de Waterloo, Canada	Rapporteuse
M. Lionel Seinturier	Professeur à l'Université de Lille, France	Examinateur
M. Pierre Sens	Professeur à Sorbonne Université, France	Rapporteur

Abstract

Information and Communication Technology is increasingly spanning worldwide, with digital services hosted all around the globe and often belonging to complex systems, utilizing many other services themselves. And ICT is currently estimated to grow at a faster pace than any other sector in terms of energy consumption. Yet, in the current context, it seems urgent to bend this alarming curve. Therefore, despite the ICT systems' complexity, understanding how they consume energy is important in order to hunt wasted Joules.

My research activities since October 2012 have focused on large-scale distributed systems and their energy consumption with a strong focus on performing real experiments and designing simulation models.

This work covers three main axes: understanding energy consumption, improving energy efficiency and greening distributed infrastructures. The first axis concerns the measurement, modeling and simulation of the energy consumption of distributed infrastructures. The second axis focuses on tackling the non-power proportionality of computing resources, redesigning cloud infrastructures and involving users in energy saving policies. The third axis provides contributions to enable data centers to partially rely on renewable energy sources, to enable distributed clouds to cooperate with Smart Grids for a better self-consumption of on-site renewable energy, and to analyze the impact of distributed computing systems piloting Smart Grids.

Résumé

Internet s'étend de plus en plus dans le monde entier, avec des services numériques hébergés tout autour du globe et utilisant souvent eux-mêmes de nombreux autres services. La complexité de ces systèmes rend leur consommation d'énergie difficile à analyser. Pourtant, comprendre leur comportement est indispensable pour optimiser cette consommation et chasser les joules gaspillés. On estime actuellement que la consommation d'énergie des technologies de l'information et de la communication croît à un rythme plus rapide que tout autre secteur, il semble urgent d'infléchir cette courbe alarmante.

Ce manuscrit donne un aperçu partiel de mes activités de recherche depuis octobre 2012. Elles concernent les systèmes distribués à grande échelle et leur consommation d'énergie avec un fort accent sur la réalisation d'expérimentations réelles et la conception de modèles de simulation.

La première partie de ce manuscrit présente mes travaux sur la mesure, la modélisation et la simulation de la consommation énergétique des infrastructures distribuées. La deuxième partie se concentre sur mes efforts pour lutter contre la non-proportionnalité des ressources informatiques, pour repenser les infrastructures cloud et pour impliquer les utilisateurs dans les politiques d'économies d"énergie. La troisième partie donne un aperçu de mes contributions pour permettre aux centres de données de s'appuyer partiellement sur des sources d'énergie renouvelables, pour permettre aux clouds distribués de coopérer avec les Smart Grids pour une meilleure autoconsommation en provenance de sources d'énergie renouvelables et pour analyser l'impact des systèmes informatiques distribués pilotant les Smart Grids.

Contents

List of Figures

List of Tables

A word after a word
after a word is power.
Margaret Atwood

I
Introduction

This manuscript gives a partial overview of my research activities since my recruitment as a permanent research scientist at CNRS in October 2012. This work has been conducted at IRISA laboratory in the Myriads team.

This chapter introduces the general context and presents the scientific challenges addressed in this manuscript.

I.A Context

In 2018, Information and Communication Technology (ICT) was estimated to absorb around 3% of the global energy consumption [The18]. This consumption is currently estimated to grow at a rate of 9% per year, faster than any other sector [The18]. But, in a century already well underway, and that is expected to experience the lack of fossil energy resources, it seems urgent to halt this alarming growth. I think the first step towards solutions to bend this energy curve lies in understanding the root causes of the consumption growth.

Several factors could explain this growth: population increase, fast emergence of numerous bandwidth-hungry applications and new ICT devices connected to Internet: sensors, smartphones, tablets, vehicles, smart appliances, etc. But, ICT and computer science benefit largely to other sectors, such as agriculture, manufacturing, energy, transport, construction, culture, health, etc. ICT and computer science offer unique means to model complex interactions, to treat massive heterogeneous data and to provide real-time optimized decisions, to name a few contributions. As they have the power to improve everyday life of billions of people, careful attention should be paid on avoiding unneeded energy waste on this road to digital transformation.

Since world population grows and a higher percentage gets access to Internet, the multiplication of ICT devices seems unavoidable. This increase translates into more energy needed to manufacture and utilize ICT devices. On the other side, more devices also generate more network traffic, and the advances in telecommunication technologies amplify this growth by providing increasing bandwidth capabilities. Since the beginning of Internet, the number of ICT devices and the exchanged traffic are expected to grow every year. Since 2012 and the beginning of the work described in this manuscript, this trend has not abated.

In May 2013, a report by Cisco stated that the global IP traffic will increase threefold over the 2012-2017 period [Cis13]. According to the 2019 Cisco's report, the 2013 prediction was right [Cis19]. And this trend will go on, according to the same report: the annual global IP traffic will increase threefold over the 2017-2022 period [Cis19]. During the same period, the busiest hour in a day in terms of Internet traffic will grow even more rapidly than average Internet traffic [Cis19]. This traffic growth will be absorbed by the Internet service providers through an increase of their communication infrastructure, and consequently more devices. The increase of network traffic and of number of ICT devices are indeed tightly coupled.

More users, more traffic, more devices, and consequently more energy. Figure I.1 offers a closer look on the 2012-2022 period and summarizes five metrics from various sources:

- the worldwide annual energy consumption due to the ICT sector (including manufacturing, transport and usage phases), based on [The18];
- the averaged global IP traffic per month, based on [Cis13, Cis14, Cis15, Cis16, Cis17, Cis19];
- the global number of IP devices, based on [Cis13, Cis14, Cis15, Cis16, Cis17, Cis19];
- the worldwide population, based on [Wor19];
- the number of Internet users worldwide, based on [Uni18, Sta18].

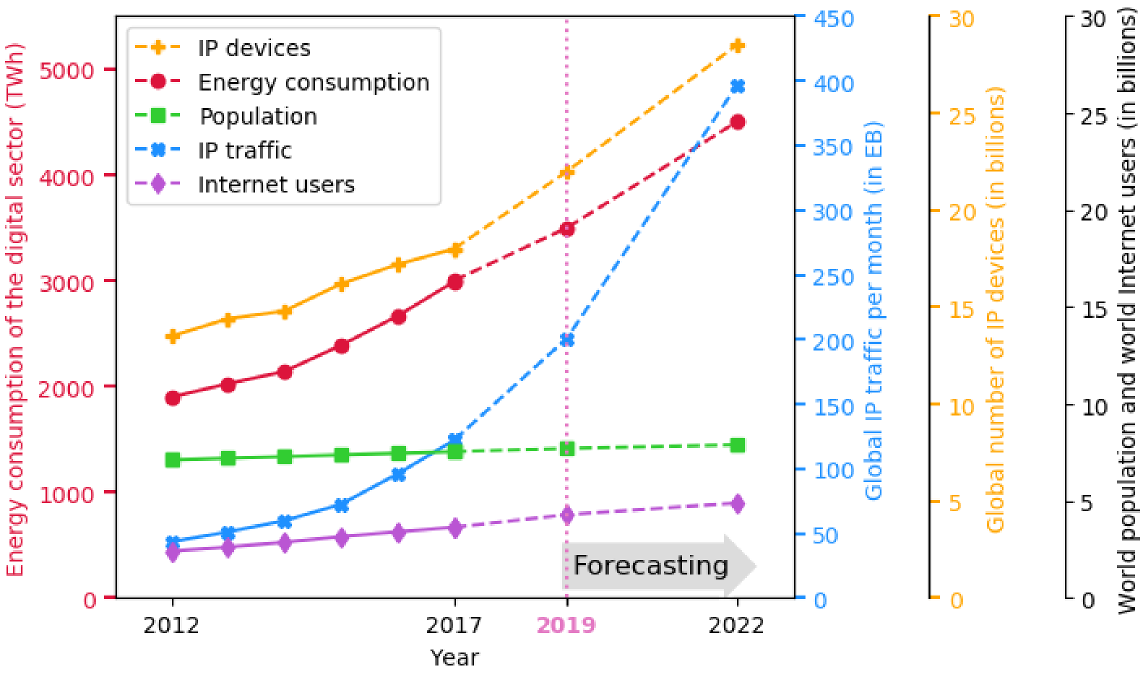


Figure I.1 – Global estimation of the energy consumption due to ICT along with the IP traffic per month, the number of IP devices, the population and the number of Internet users.

The values for 2012 to 2017 are estimates grounded on measured data, while the values for 2019 and 2022 are predictions. Several observations can be made from this figure:

- The number of IP devices grows faster than both the number of connected users and the worldwide population: there is more and more devices on the Internet.
- The global monthly traffic grows much faster than the number of IP devices: each user and each device produce more and more traffic.
- According to the prediction, the global energy consumption due to ICT will more than double between 2012 and 2022: ICT continues to be more energy-hungry at an alarming pace.
- The global energy consumption seems more related to the number of IP devices than to the IP traffic.

Intuitively, if the usage phase was dominant for all ICT devices and if they were perfectly power-proportional (i.e. their power consumption would depend only on their utilization, and they would not consume power when not used), the energy consumption curve should follow the traffic curve. Yet, the latter observation contradicts this intuision, and several reasons could explain this: (1) the usage phase is negligible in terms of energy consumption compared to the manufacturing phase; or (2) traffic's influence on the energy consumption of ICT devices is negligible (i.e. they are really far from being power-proportional); or (3) the energy efficiency of ICT devices is fully increasing along with the traffic (i.e. the improvements brought by each new generation of computing and communication technology absorbs the capacity increase: it provides more bandwidth and more computation while consuming the same); or (4) the answer lies in between these three reasons; or (5) there is a missing factor that we did not spot, nor plot in this graph. Finding the answer is complicated because of the intrinsic system's complexity.

As Internet usage has grown, IT systems have become more elaborate in order to meet user demand, availability and quality of service requirements. Even a simple electronic mail service, one of the oldest Internet application, requires now many software and hardware components to operate and to ensure the by-default quality-of-service expected by users in 2019. To provide robustness, low-latency, and large storage and computing power, ICT infrastructures have gained weight: servers have gathered in many super-sized data centers, networks have pursued their expansion in terms of coverage and bandwidth, and software have followed the same way by expanding in size and interdependence with multiple other software pieces.

Internet now spans worldwide, with digital services hosted all around the globe and often utilizing many other services themselves. The systems' complexity makes its energy consumption quite difficult to analyze. Yet, understanding its behavior is mandatory to optimize it and to hunt wasted Joules. I started to work on this problem in 2008 during my master internship on *Energy-aware frameworks for high-performance data transport and computations in large-scale distributed systems*. I pursued this research direction during my PhD thesis entitled: *An Energy-Efficient Reservation Framework for Large-Scale Distributed Systems*, and defended in September 2011. I would like to start this manuscript with the state-of-the-art chapter of my thesis that was published in:

"A Survey on Techniques for Improving the Energy Efficiency of Large Scale Distributed Systems", Anne-Cécile Orgerie, Marcos Dias de Assunção and Laurent Lefèvre, *ACM Computing Surveys*, ACM, volume 46, issue 4, pages 47:1-47:31, April 2014.

This survey paper ends with the following paragraph:

> *"After exploring studies and models for estimating the energy consumption of these resources, we presented a classification of existing solutions and research work. Although many research directions have been studied to save energy, several key problems remain open. Are virtualization and Cloud computing the panacea for saving energy? Which architecture is the most energy efficient: centralized, totally distributed, or something in-between? What is the best way to explore the trade-offs between energy and performance? How is energy proportionality reached? One of the main leverages to reduce the electric bill and the carbon footprint of IT infrastructure is to increase the energy awareness of users and providers."*

An entire research program!

I.B Challenges

As illustrated by Figure I.1, reducing the energy consumption of ICT infrastructures is critical. To my opinion, this necessitates first to understand in depth how this energy is consumed, despite the intrinsic complexity of the system. Several options can then be explored to reduce the energy consumption and the carbon footprint of ICT infrastructures. Among the wide research avenues sketched above and at the end of the survey paper, I strode along the following ones:

- Understanding the energy consumption of distributed infrastructures,
- Improving the energy efficiency of distributed infrastructures,
- Increasing the part of renewable energy in the electricity mix powering distributed infrastructures.

I.C Organization of the manuscript

This manuscript highlights the main results related to my work since 2012. Detailed experiments and algorithms can be found in the original papers. I give here a non-exhaustive overview in order to focus on the main research directions I explored, and provide a coherent and concise structure to this document. It should be highlighted that my work focuses on the use phase; therefore, the equipment manufacturing phase is outside the scope of this document.

Chapter II presents my work on measuring, modeling, and simulating the energy consumption of distributed infrastructures.

Chapter III focuses on my efforts to fight against the non-power proportionality of computing resources, to redesign cloud infrastructures, and to involve users in energy saving.

Chapter IV provides an overview of my contributions to enable data centers to partially rely on renewable energy sources, to allow distributed Clouds to cooperate with Smart Grids for a better self-consumption of on-site renewable energy, and to analyze the impact of the distributed computing systems piloting the Smart Grids.

Finally, Chapter V concludes this manuscript and sketches future research directions that I would like to pursue. Later. If I am habilitated to supervise PhD students.

For each part of the manuscript, I indicate the related publications in an attempt to thank my colleagues with whom we embarked on these adventures, and to provide the reader with pointers to more detailed versions of the work. In the case of publications with colleagues working outside of France (regardless of their nationality), I add the corresponding country flag to highlight international collaborations. In all cases, the file icon is a link to the pdf version hosted on hal[1], the French open archive.

[1] `https://hal.archives-ouvertes.fr`

The beginning of knowledge
is the discovery of something
we do not understand.

Frank Herbert

II
Understanding the energy consumption of distributed infrastructures

II.A Introduction to energy monitoring

The maturity of virtualization techniques has enabled the emergence of complex virtualized cloud infrastructures capable of rapidly deploying and reconfiguring virtual resources. These infrastructures provide users with resources that are dynamic, reliable and tailored to their needs. By benefiting from economies of scale, these distributed infrastructures can efficiently manage their resources and offer virtually unlimited storage and computing capabilities, while minimizing costs for users.

However, the rapid expansion of these infrastructures leads to a disturbing and uncontrolled increase in their electricity consumption. In order to limit this consumption, it is first necessary to understand how these infrastructures consume energy and thus to monitor them. This fine energy monitoring is a challenge for cloud infrastructures, because of the size of these systems and the stacking of their software layers. We addressed this challenge in the focus group on *metrics, monitoring, instrumentation and profiling* in the European COST Action NESUS[1]. Our results have been published in:

"**Energy Monitoring as an Essential Building Block Towards Sustainable Ultrascale Systems**", Francisco Almeida, Marcos Dias de Assunção, Jorge Barbosa, Vicente Blanco, Ivona Brandic, Georges Da Costa, Manuel F. Dolz, Anne C. Elster, Mateusz Jarus, Helen Karatza, Laurent Lefèvre, Ilias Mavridis, Ariel Oleksiak, Anne-Cécile Orgerie and Jean-Marc Pierson, *Sustainable Computing: Informatics and Systems*, Elsevier, volume 17, pages 27-42, March 2018.

In particular, we identified that current monitoring systems still encounter the following difficulties:

- scalability: monitoring should be able to deal with a wide range of metrics with per-second granularity over large clusters;
- monitoring overhead: monitoring solutions should not incur significant costs on the monitored system;
- data management: energy monitoring can become a so-called big-data system as not only power consumption information is collected, but also information on resource usage and other variables with which information is correlated during analysis;
- scalable architectures: monitoring solutions should be able to scale horizontally, allowing administrators to add more capacity on demand;

[1]NESUS: Network for Sustainable Ultrascale Computing, `http://www.nesus.eu`, COST (European Cooperation in Science and Technology) Action IC1305 (2014-2018)

- power consumption models: the use of power models can help reduce the number of required monitored devices and sample rates when resources across a cluster are statistically similar.

Given suitable monitoring and test-bed infrastructures, studying the consumption of distributed infrastructures can reveal important sources of energy waste. I believe that searching for these sources is a good way to find approaches to reduce the ICT energy consumption. The methodology that I followed consists in measuring the energy consumption, deriving models from these measurements and implementing these models into simulation tools. The models also serve to build usable metrics for cloud providers and users in order to perform platform sizing or energy cost awarding for instance. As for the simulation tools, they are employed to try and validate new algorithms and architectures, and evaluate whether they save energy or not.

This chapter presents my work on measuring (Section II.B), modeling (Section II.C), and simulating (Section II.D) the energy consumption of distributed infrastructures. Finally, Section II.E presents perspectives on this work.

II.B Energy consumption of Cloud infrastructures

Since the appearance of the Internet, the vast majority of ICT equipment relies on distributed infrastructures: routers and servers, linking end-user devices and providing online services. Current Internet services massively lean on top of Cloud infrastructures: virtualized computing resources geographically distributed in data centers connected through telecommunication networks. In 2017, datacenters consumed 593 TWh [The18], which represented 2.7% of the worldwide energy consumption [Ene18]. In this section, we investigate the energy consumption of Cloud infrastructures, and in particular, we scrutinize their data centers.

From a software point of view, in Clouds, hardware resources are virtualized and monitored by several layers of middleware and software. Typical Cloud models include several stackable layers: Infrastructure-as-a-Service (IaaS) layer on top of virtualization technologies and that delivers virtual computing resources; Platform-as-a-Service (PaaS) layer on top of IaaS layer that delivers a development and execution environment for user's applications; and Software-as-a-Service (SaaS) layer that provides an access to ready-to-use applications.

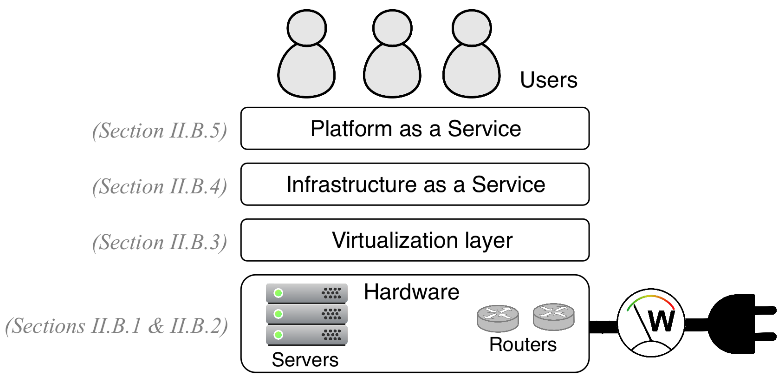


Figure II.1 – Hardware and software layers involved in the management of Cloud users and resources

Figure II.1 shows a schematic view of the software layers a Cloud user goes through in order to access hardware resources. In the following, we will examine each part in a bottom-up approach: from the hardware resources, namely servers (Section II.B.1) and network devices (Section II.B.2), that are close to the monitoring infrastructure and the wattmeters if any, through the virtualization (Section II.B.3) and IaaS layers (Section II.B.4) and finally to the PaaS layer to which the user can directly access (Section II.B.5).

II.B.1 Computing part: the servers

The work presented hereafter has been published in:

> **"Predicting the Energy-Consumption of MPI Applications at Scale Using Only a Single Node"**, Franz C. Heinrich, Tom Cornebize, Augustin Degomme, Arnaud Legrand, Alexandra Carpen-Amarie, Sascha Hunold, Anne-Cécile Orgerie and Martin Quinson *IEEE Cluster Conference*, Hawaii, USA, pages 92-102, September 2017.

Power consumption of servers is often modeled as the sum of two separate parts [59]: a static part that represents the consumption when the server is powered on but idle; and a dynamic part, which is linear in the server utilization and depends on the CPU frequency and the nature of the computational workload (e.g. computation vs. memory intensive, provided such characterization can be done). To assess these properties, we conducted an experimental campaign on the Grid'5000 infrastructure [BCAC+13], a French testbed for experiment-driven research[2]. This testbed provides bare-metal deployments for users with root privileges, allowing access to fully configurable servers. In 2020, Grid'5000 comprises 8 sites, overall accounting for 34 homogeneous clusters with a total of 774 servers. In particular, we employ the taurus cluster due to the availability of accurate hardware wattmeters. The monitoring ensures a sampling rate for each entire machine of 1Hz with an accuracy of 0.125 Watts. The taurus cluster is composed of 16 homogeneous nodes; each node consists of 2 Intel Xeon E5-2630 CPUs with 6 physical cores per CPU and 32 GB of RAM.

For this evaluation, we use three MPI applications. The first two originate from the MPI NAS Parallel Benchmark suite (v3.3). The NAS EP benchmark performs independent computations in an embarrassingly parallel way. The NAS LU benchmark performs a Lower-Upper decomposition using the Gauss-Seidel method. Finally, we selected the HPL benchmark (v2.2) as it is commonly used to rank supercomputers both in the TOP500 and in the Green500 [Top19].

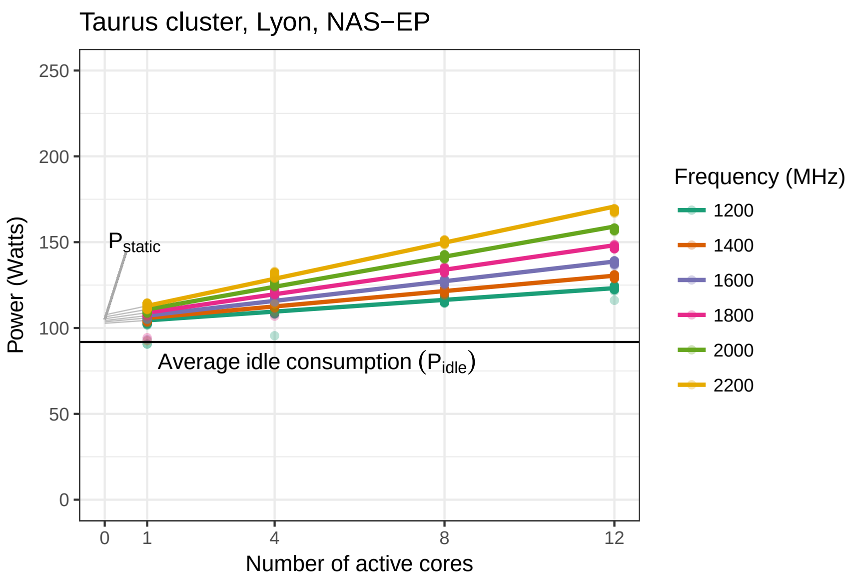


Figure II.2 – Power consumption on `taurus-8` when running NAS-EP, class C, varying the frequency and the number of active cores.

Figure II.2 illustrates the linearity in load of power consumption for the NAS EP benchmark on `taurus-8`. Yet, this power profile is application-dependent: the nature of the computational workload impacts the power consumption. In other words, two applications using 100% of the CPU may have different power consumption as illustrated by Figure II.3. This figure shows that

[2] Grid'5000 `https://www.grid5000.fr`

the power consumption differs when running NAS-EP, NAS-LU or HPL, although all the three benchmarks fully utilize all the cores of the servers.

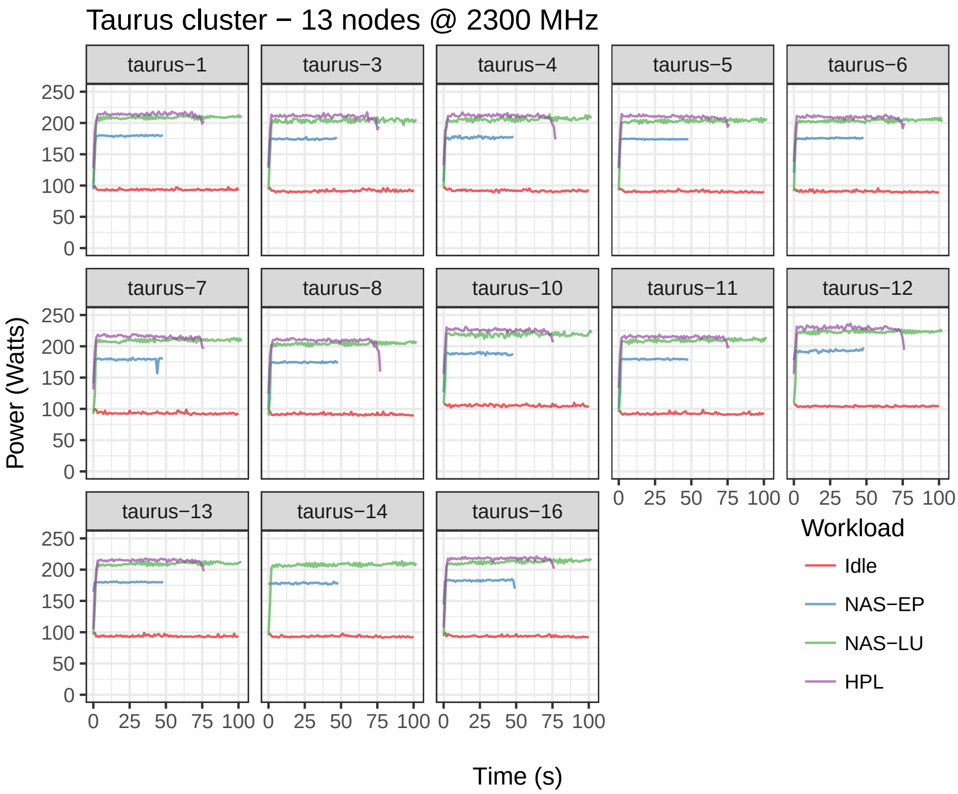


Figure II.3 – Power consumption over time when running NAS-EP, NAS-LU, HPL or idling (with 12 active cores and the frequency set to 2300MHz).

Note that the measurements of Figure II.2 show that it is generally safe to assume P_{static} is independent of the frequency, but that it should not be confused with the fully idle power consumption P_{idle}. This can be explained by the fact that when a CPU goes fully idle, it can enter a deeper sleep mode, which further reduces its power consumption.

Figure II.3 also highlights the heterogeneity in power consumption among identical servers running the same application or even idling. Interestingly, this experimental work spanned over several years and we did several complete measurement campaigns on the taurus cluster at different time periods. We were for instance able to compare the idle power consumption along time on two different dates (May 2014 and October 2016) for various nodes. We observed significant differences with an increase of the power consumption for some servers (more than 10%), and a decrease on others.

Consequently, every power state (static, idle, off) and every new application requires a specific series of potentially tedious measurements. However, in our opinion, they can hardly be avoided. Computers have become increasingly complex and even minor modifications to the setup can have major impact on performance [MDHS09]. In this experimental study, we made an inventory of all parameters that may influence the behavior of the system, in terms of both speed and power consumption (see Figure II.4). We identified these parameters as the principal ones: every experimenter should track them so that a faithful decision can be made whether or not the system requires to be re-calibrated.

The first category is related to the hardware at hand. The second category of factors is actually related to when the system is measured. Computers are indeed quite sensitive to temperature and so can the temperature of the machine room affect the speed of processors, their power consumption and even sometimes their clock drift [All87]. The following categories are related to the operating

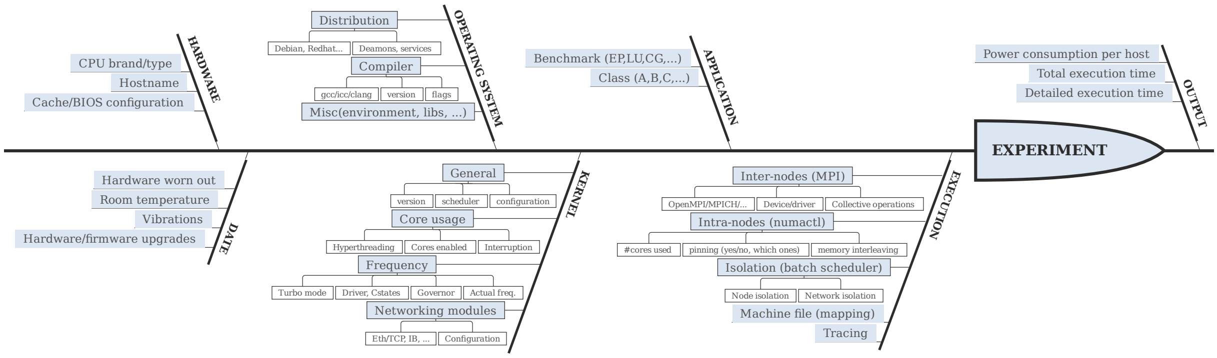


Figure II.4 – Causal diagram associated with the performance of an HPC system.

system, the kernel configuration, the application itself, and the runtime.

Although these results were published in September 2017 [44] in the context of the Hac Specis project[3], our first discussion on modeling the energy consumption of servers started in December 2008 between the last two authors of the final paper. Our first experimental campaign was done in late 2009, after the installation of the Grid'5000 wattmeters. The final paper only reflects the last part of this work: the successful model and its experimental validation. Many fails on the way were not documented, nor published. They contributed to the experience that we gained on performing reproducible energy measurements on distributed infrastructures.

Our pugnacity has been rewarded by the confidence that we now have in our results and in our experimental methodology. I think more attention should be paid to these methodological aspects as they alone can guarantee trustworthy scientific results. Figure II.4 is part of a research report – the initial version of the final (rejected) paper – that summarizes our key findings on the experimental methodology. Both papers (the report and the published paper), along with the experimental results, are available at `https://gitlab.inria.fr/fheinric/paper-simgrid-energy`.

II.B.2 Wired network devices

Along with servers, Cloud infrastructures heavily employ wired network devices (routers and switches), either internally in the data centers or externally to connect users to the Cloud. While the energy consumption models for servers mainly focus on CPU utilization, for wired network devices, they utilize network traffic. The work presented hereafter was done during the Bachelor internship of Timothée Haudebourg and got the best paper award of ICA3PP 2017:

> **"On the Energy Efficiency of Sleeping and Rate Adaptation for Network Devices"**, Timothée Haudebourg and Anne-Cécile Orgerie, *ICA3PP: International Conference on Algorithms and Architectures for Parallel Processing*, Helsinki, Finland, pages 132-146, August 2017, *Best Paper Award.*

Network infrastructures present high device redundancy and low utilization upper bounds for fault-tolerance and security purposes. This directly impacts their energy consumption as it requires more devices. Moreover, network devices, such as routers and switches, are power-hungry even when they are little or not used [FFKR15, PP18]. Although experimental measurements on real infrastructures constitute the panacea to better understand energy consumption, they are not always feasible. In the case of networking equipment, it requires to control all the incoming and outgoing traffic (including ssh connections for instance) and to have high-precision wattmeters because the involved power values are small: the power consumption of a 1 Gbps Ethernet port is around 1 Watt [FFKR15, NHQ+15].

[3] Inria project lab on High-performance Application and Computers: Studying PErformance and Correctness In Simulation (2016 - 2020) `http://hacspecis.gforge.inria.fr`

Experimental studies from the literature show that, as for servers, the energy consumption of network devices comprises a fixed part (the idle consumption) and a dynamic part. This latter usually depends on the number of processed packets and the number of processed bytes [SVZR11, NHQ+15].

The ideal power-proportionality has still not been reached by device manufacturers as the idle power consumption on network devices can still reach 85 to 95% of their maximal power consumption [NHQ+15]. These observations have led to the proposition of various solutions to save energy in wired networks. Approaches found in literature can be categorized into two categories, both exploiting the lower charge periods to either put to sleep some hardware elements (sleeping) or adapt the network rate to the actual traffic level (rate adaptation).

The emblematic sleeping solution proposes a standardized Low Power Idle (LPI) mode [CRN+10] (norm IEEE 802.3az). The basic idea of this Energy-Efficient Ethernet (EEE) standard consists in sending packets as fast as possible and entering a low-power idle state when there is no data to transmit. The first network devices implementing this capability have appeared on mass market in 2013. Packet coalescing can be used to improve LPI performances at the cost of a slight latency increase [CM16]. As for rate adaptation, the most famous implementation is Adaptive Link Rate (ALR) which has been proposed in 2005 [GCN05]. It follows the idea of the Dynamic Voltage Frequency Scaling (DVFS) for CPUs adapted to the network device port rates. When full speed is not needed, a lower rate is negotiated between the network ports sharing a common link, thus incurring less power consumption [BCN06].

While these two techniques pursue a common goal, they adopt radically different approaches. The only study comparing both approaches that we found in literature proposes a theoretical comparison based on models of sleeping and rate adaptation general techniques [NPI+08]. In particular, as this study was published in 2008, before the adoption of IEEE 802.3az, their sleeping model employs values differing by an order of magnitude from the one actually implemented in Low Power Idle (for the switching time for instance).

We conducted a simulation-based comparison relying on an implementation of the two existing protocols (LPI and ALR) under various traffic conditions, using the network packet-level ns-3 simulator [ns3]. Contrarily to previous work, we showed that LPI has a clear advantage in terms of energy savings compared to ALR, and an even larger advantage on QoS for most of the traffic scenarios. Our results also indicate that combining both protocols, LPI and ALR, reduces the energy saving dependence to packet coalescing. But, at the same time, it hugely impacts the latency and jitter, thus making LPI alone more suitable. Consequently, we concluded that, with the current state-of-the-art hardware, ALR should stop being considered as a suitable solution by the community.

This study gave us a fine understanding of the energy consumption of wired network equipment that are deployed in Cloud data centers and Internet Service Provider (ISP) infrastructures. The last experiment of the paper concerns indeed the simulation of an Italian ISP network (detailed in [CMN09]). We conducted another study to analyze the energy consumption induced on network devices by virtual machine migration inside data centers. This work was done during the post-doc of Bogdan Cornea, and has been published in:

"**Studying the energy consumption of data transfers in Clouds: the Ecofen approach**", Bogdan Cornea, Anne-Cécile Orgerie and Laurent Lefèvre, *CloudNet: IEEE International Conference on Cloud Networking*, Luxembourg, Luxembourg, pages 143-148, October 2014.

II.B.3 Virtualization layer

Combining computing and networking devices, Cloud Computing has become one of the main technologies in the Internet, particularly at an Infrastructure-as-a-Service (IaaS) level, due to its virtualization of resources. One of the main objectives of virtualization is that several clients can execute their services on the same physical server, keeping these services isolated from each other. This virtualization layer, either directly above hardware devices or on top of an operating system layer, hides the actual resource utilization from the user, and consequently, their energy consumption becomes even more complex to determine. The work presented hereafter has been published in:

"**Comparative Experimental Analysis of the Quality-of-Service and Energy-Efficiency of VMs and Containers' Consolidation for Cloud Applications**", Ismael Cuadrado Cordero, Anne-Cécile Orgerie and Jean-Marc Menaud, *SoftCOM: International Conference on Software, Telecommunications and Computer Networks*, Split, Croatia, pages 1-6, September 2017.

Virtualization techniques advocate for consolidation that allows to gather several virtual environments on the same server to optimize resources. Currently, virtualization of resources is done mainly through two technologies: Virtual Machines (VMs) and containers. VMs emulate all the functionalities of a physical machine, while containers are instances running all on the host Operating System's kernel. Containers are a more lightweight virtualization technology than VMs, and have seen a growing popularity in the last years.

We studied the impact that consolidating multiple virtualized services on the same server has on quality of service and energy consumption. The service to evaluate is a LAMP stack (Linux-Apache-MySQL-PHP), virtualized in the same host server using different technologies, as it is a very extended archetypal model of existing web services. LAMP is named after the four open-source components from which is formed: Linux (OS), Apache (HTTP Server), MySQL (database), and PHP (programming language). Each service runs the web service benchmark RUBiS that simulates multiple concurrent users in an on-line auction market [Ric]. Each experiment simulates an increment of users connecting to a single service (from 0 to 3000 users).

The experiments uses the taurus cluster of Grid'5000 [BCAC+13]. The cluster has been divided into one server and several clients. For the VM technology, we employ KVM and Docker for the containers. Successive experiments deploy respectively 1, 2, 5, 10, 15, 20 and 25 different services in the same server, the number of clients is scaled accordingly to stress the service. For each couple of experiments (VMs and containers), the same server has been used to avoid measurements' disruptions due to heterogeneity, that may have appeared in the cluster's lifetime.

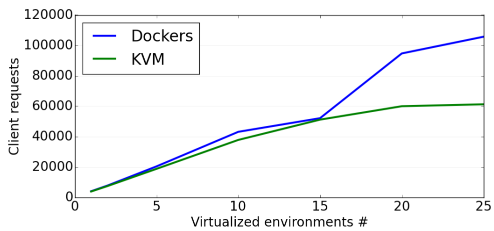


Figure II.5 – Evolution of client requests successfully managed by the server over an increasing number of services

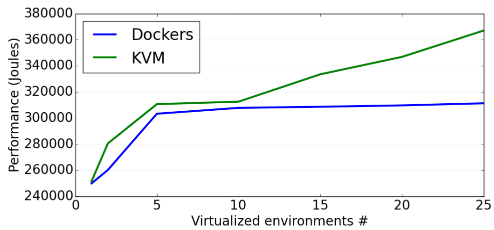


Figure II.6 – Evolution of energy consumption over an increasing number of services

Figure II.5 shows the evolution on requests successfully managed by the server when the number of services running on it increases. Docker outperforms KVM both in quality of service and energy efficiency. According to our measurements, Docker allows running up to a 21% more services than KVM, when setting a maximum latency of 3,000 ms for answering the clients requests. In this configuration, Docker offers this service while using a 11.33% less energy than KVM. At a datacenter level, the same computation could run using less servers and less energy per server, accounting for a total of a 28% energy savings inside the datacenter.

These experiments show that software layers managing computing resources can have an accountable impact in terms of both server utilization and energy consumption. Virtualization layer being only the first layer above the hardware resources, this phenomenon should increase in the upper layers.

II.B.4 Infrastructure as a Service layer

Directly above the virtualization layer as depicted in Figure II.1, the Infrastructure-as-a-Service (IaaS) layer provides VMs to users. The work presented hereafter has been published in:

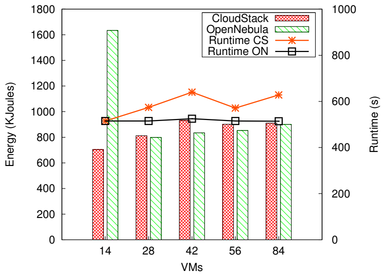
"**Experimental Study on the Energy Consumption in IaaS Cloud Environments**", Alexandra Carpen-Amarie, Anne-Cécile Orgerie and Christine Morin, *IEEE/ACM International Conference on Utility and Cloud Computing (UCC)*, Dresden, Germany, pages 42-49, December 2013.

"**An experiment-driven energy consumption model for virtual machine management systems**", Mar Callau-Zori, Lavinia Samoila, Anne-Cécile Orgerie and Guillaume Pierre, *Sustainable Computing: Informatics and Systems*, Elsevier, volume 18, pages 163-174, June 2018.

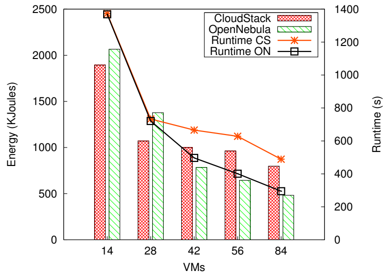
We explored the energy consumption patterns of IaaS cloud environments under various synthetic and real application workloads. We focused on the two main open-source IaaS frameworks at the time of this work (2013): Apache CloudStack and OpenNebula. We deployed each of the two cloud frameworks on 15 nodes of the taurus cluster of Grid'5000, dividing them into one cloud frontend node and 14 compute nodes, using KVM as the VM hypervisor for both clouds. As the application benchmark, we used three distributed applications from the Hadoop MapReduce implementation [Had]: Pi that estimates the value of Pi using a quasi-Monte Carlo method, Grep that is designed to search for a specific pattern in very large files, and Sort that is devised to sort key/value pairs in a distributed fashion.

Figure II.7 presents the execution time of each application on the right side Y-axis, when increasing the number of VMs processing the same workload. The results show that the job completion time decreases as the virtual cluster is expanded, for the two IO-bound applications (Grep and Sort). The *Pi* application exhibits a different behavior when we increase the number of CPUs and adjust the number of Hadoop mappers accordingly. A larger number of available mappers is equivalent to more processing power that increases the accuracy of the result (i.e., the number of decimals computed for Pi). In this case, the runtime is not a measure of the application performance, but it rather emphasizes the scalability of the VM cluster. In the case of OpenNebula, the runtime for *Pi* is constant regardless of the number of VMs, as a consequence of its round-robin allocation strategy, allowing the framework to achieve similar performance for all its VMs. On the other hand, CloudStack has a different VM management strategy (i.e. random with overcommit), which often leads to less than optimal VM distribution across compute nodes and consequently, to higher execution times.

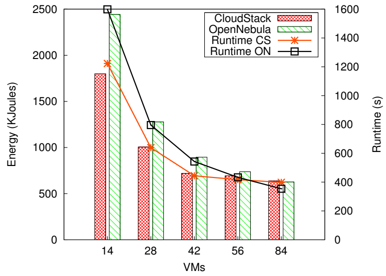
As far as the *Grep* and *Sort* applications are concerned, the runtime drops as we deploy more 2-core VMs. The *Sort* application produces similar results as *Grep*, yielding a substantial performance gain when the virtual cluster size or CPU capacity per VM are augmented to process the same amount of data. *Sort* is a representative data-intensive application, for which most of the execution time accounts for data reading and writing. Unlike *Grep*, which spends a significant percentage of its runtime for processing data, *Sort* is mostly impacted by the disk and network capabilities. For this reason, Figure II.7(c) displays less steep runtime gain than the equivalent *Grep* results. From the IaaS point of view, OpenNebula ensures a limited performance gain for the *Grep* application over CloudStack. This advantage is only noticeable for a large number of VMs, due to OpenNebula's allocation strategy that prevents compute nodes overloading and thus, reduces disk and network contention. The *Sort* application shows a clear advantage of CloudStack over OpenNebula for the same reason.



(a) Distributed Pi (CPU-intensive).



(b) Distributed Grep (CPU- and IO-intensive).



(c) Distributed Sort (IO-intensive).

Figure II.7 – Execution runtime and total energy consumed by each cloud environment.

These experiments show how different workload types and configuration decisions affect the energy profile of each cloud. By considering the energy consumption of the entire cloud, our evaluations provide valuable insights on cloud computing potential to save energy. For instance, while data-intensive applications benefit from overcommit strategies, a round-robin allocation ensures better execution time and energy consumption for compute-intensive applications.

We pursued this work by studying OpenStack [Ope], another open-source Cloud middleware that gained a lot of attention in the recent years. We investigated the energy consumption of VMs management operations, such as VM placement, VM start up and VM migration. An illustrative example is provided on Figure II.8 for the VM start up case on a server already hosting m VMs. These measurements, performed on an Orion server from Grid'5000, exhibit the variability that we encountered when dealing with OpenStack, which is a complete Cloud stack relying on numerous service components interacting among them for each VM management atomic operation. Figure II.8 also shows the additional energy consumption of starting an idle VM on an OpenStack compute node that is twofold: during the boot phase and afterwards.

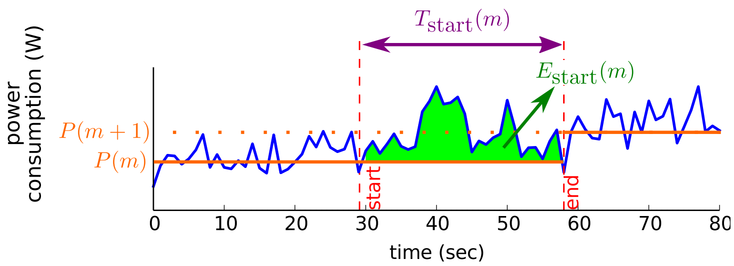


Figure II.8 – Power consumption when starting a VM.

Although this section (Section II.B.4) is small in number of lines, it represents a considerable investment in number of experimentations. The first part comparing CloudStack and OpenNebula was performed during the post-doc of Alexandra Carpen-Amarie and accounted for more than 4,423 days of core computation on Grid'5000 [4]. The second part on basic OpenStack operations was performed during the post-doc of Mar Callau-Zori and the internship of Lavinia Samoila and accounted for more than 13,497 days of core computation on Grid'5000 [5].

We tried to further investigate in this direction with OpenStack [Ope] during the post-doc of Anthony Simonet. Our idea was to measure the energy consumption of the various middleware components (i.e. network service, storage service, virtual machine controller, etc.) in order to identify the most consuming ones. We deployed each service on separate nodes of Grid'5000 and

[4] See oarstat for the accurate value: `oarstat -f -u acarpena --accounting "2013-01-01, 2013-08-31"`

[5] See oarstat: `oarstat -f -u mcallauzori --accounting "2014-05-01, 2015-08-31"` and `oarstat -f -u lsamoila --accounting "2014-03-01, 2014-12-31"`

measured their energy consumption while performing basic cloud operations such as listing the available compute nodes, booting virtual machines and creating virtual networks. Yet, this work was unsuccessful as we obtained plots exhibiting high variations that we could not correlate with our activity monitoring reports. This work highlighted the experimental issues faced in the context of complex distributed software:

- there is only little control on the reproducibility of the experiments due to the complexity of middleware deployments,
- the Grid'5000 wattmeters provided insufficient precision at this time especially for the sampling frequency (1Hz),
- the middleware complexity is too high to be efficiently monitored without altering the measurements due to the intricacy of the numerous software components and their fault-tolerance mechanisms leading to unpredictable behaviors.

II.B.5 Platform as a Service layer

On top of the IaaS layer stands the Platform-as-a-Service (PaaS) layer that delivers a development and execution environment for user's applications. This layer experiences growing success [Rig15], however, few research works have been conducted on the possible energy optimization that can be done at this cloud service layer. The work presented hereafter has been published in:

"**An Experimental Analysis of PaaS Users Parameters on Applications Energy Consumption**", David Guyon, Anne-Cécile Orgerie and Christine Morin, *IC2E: IEEE International Conference on Cloud Engineering*, Orlando, USA, pages 170-176, April 2018.

A PaaS user cannot interact with the computing resources and the underlying operating system directly. However, users have access to parameters in order to control the execution of their applications:

- software stack to use depending on the programming language of the application
- database management system to store the data
- software versions for running the application and the database

In this work, we studied the impact of PaaS level users' decisions on the energy consumption. We focus here on web applications and target the software parameters offered by PaaS clouds. Similarly to the study on VMs vs. containers, we used RUBiS [Ric], an online auction website modeled after the Internet website eBay.

The benchmark executes on top of the nova servers of Grid'5000 [BCAC+13]. Deployed in 2016, they are equipped with 16 cores from the Intel Xeon CPU E5-2620 processor, 32GB of RAM, 600GB of HDD and a 10GB Ethernet connection. The application tier and the database tier of RUBiS execute in two separate VMs. An application scenario corresponds to a VM image in which the required software stack for a specific version of RUBiS is installed and ready to use. Two scenarios are dedicated to execute the PHP version of RUBiS with respectively PHP5 and PHP7. The four remaining scenarios are for the Servlet version running with Java 7 or Java 8, and either with Tomcat 7 or Tomcat 8.

Figure II.9 displays both the dynamic energy consumed by the application tier and the database tier when we apply the workload on each application scenario (where T7J7 stands for Tomcat 7 with Java 7, and so on). The graph on the right represents the average response time of all clients to access the home page.

It shows that on average the application tier of the two PHP scenarios consumes 7.27% less energy compared to the four Java scenarios. This difference is explained by the additional cost caused by the Java Virtual Machine in the Java versions. Varying programming languages and software versions mainly impact the energy consumption of the application VM with a maximum of 12.04% more energy consumption between PHP7 and Tomcat 8 Java 7. The energy consumption

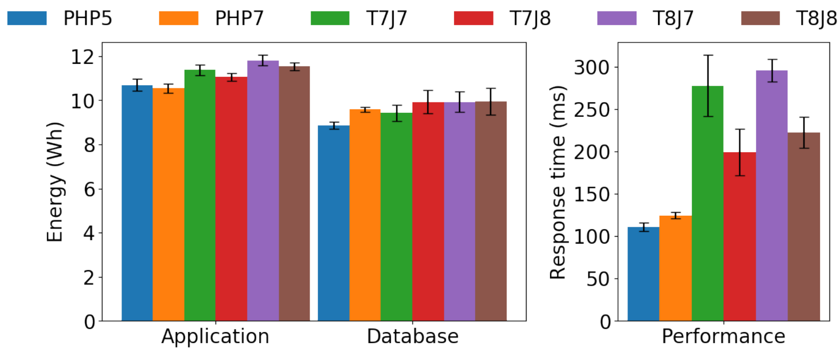


Figure II.9 – Dynamic energy consumption of application and database tiers and response time for each application scenario.

of the application VM varies also when changing the version of the software used to execute the application: we observe an increase of 4.42% when moving from Tomcat 7 to Tomcat 8.

PaaS users could use this kind of information to lower the energy consumption of their application without impacting their performances. Yet, they do not have access to this information as many layers stand in between their applications and the wattmeters (as shown in Figure II.1). For instance, the PaaS provider can use virtual machines from various IaaS providers, and consequently accessing power monitoring of hardware resources can be physically impossible. Yet, even though values, such as the dynamic energy consumption, would be available to PaaS users, it would be difficult for them to know whether a given value is good or not in absolute terms.

This concludes our section on monitoring and measuring the energy consumption of Cloud infrastructures. All the experiments conducted in this section highlight the need for understandable metrics from a user or provider point of view and multi-criteria parameters. It would be indeed easy to forget performance and quality of service to reduce energy consumption, but probably at the cost of numerous clients. On the other side, energy-aware users could be interested in accessing energy-related metrics about their utilization of Cloud infrastructures.

II.C Towards comprehensive energy metrics

From the aforementioned experimental studies, we acquired insights on how to design energy measurements. Yet, raw power measurements need to be combined with performance indicators to provide energy-efficiency values that users would demand for optimizing their application, under multiple constraints (e.g. financial cost, response time, runtime, number of satisfied users). ICT's high energy consumption starts to gain media attention and Cloud users, mostly willing to reduce their monetary costs, start to pay attention to the energy consumption of their devices and cloud applications [SNSC18]. These users necessitate adequate metrics to gain knowledge on the energy profile of their applications, and to identify energy losses that could be reduced.

For instance, in order to optimize the energy consumption of a given Cloud application, one has to estimate the impact of the connected devices periodically sending data to the Cloud servers that process them. Ideally, this impact should be quantified in terms of energy consumption or carbon emissions. It means, that from raw measurements through wattmeters attached to Cloud hardware resources, we should devise models to estimate the overall energy impact of a given user or service. In this delicate quest, one should remind that Cloud users also affect the indirect energy consumption of servers, since data centers require cooling systems, power distribution units, etc.

We addressed this issue of energy efficiency in large-scale systems in front of other quality metrics in a book chapter written within the context of the NESUS Cost Action. This chapter explores the design of metrics, analysis, frameworks and tools for energy awareness and energy efficiency. In particular, it deals with the *energy complexity*, reflecting the synergies between energy efficiency and quality of service, resilience and performance, by studying computation power, communication/data sharing power, data access power, algorithm energy consumption, etc. This chapter, not detailed in this manuscript, has been published in:

"**Energy aware ultrascale systems**", Ariel Oleksiak, Laurent Lefèvre, Pedro Alonso, Georges Da Costa, Vincenzo De Maio, Neki Frasheri, Victor M. Garcia, Joel Guerrero, Sébastien Lafond, Alexey Lastovetsky, Ravi Reddy Manumachu, Benson Muite, Anne-Cécile Orgerie, Wojciech Piatek, Jean-Marc Pierson, Radu Prodan, Patricia Stolf, Enida Sheme, Sébastien Varrette, *chapter in Ultrascale Computing Systems*, pages 127-188, IET (ISBN 978-1-785-61834-5), January 2019.

In an incremental way, starting from a data center infrastructure and a VM power consumption model, I investigated comprehensive energy metrics for Cloud providers and users. The resulting metrics and models are presented hereafter in this section, and they include Cloud infrastructure models (Section II.C.1), comprehensive VM energy models (Section II.C.2), end-to-end models for IoT devices (Section II.C.3) and CO_2 VM models (Section II.C.4).

II.C.1 Cloud infrastructures from provider point of view

Recently, the proliferation of new usages related to Internet of Things (IoT) calls for more distributed cloud architectures, relying on resources deployed across and at the edge of the network. Referred to as Fog and Edge computing infrastructures [MNY+18, MKB18], these emerging virtualized architectures aim at satisfying low latency and high bandwidth requirements expected by IoT-based applications. While there is no more debate on whether such infrastructures will be deployed, the question of their energy consumption compared to traditional cloud architectures remains open. The work presented hereafter was done during the post-doc of Ehsan Ahvar and has been published in:

"**Estimating Energy Consumption of Cloud, Fog and Edge Computing Infrastructures**", Ehsan Ahvar, Anne-Cécile Orgerie and Adrien Lebre, *IEEE Transactions on Sustainable Computing*, pages 1-12, March 2019.

In this work, we propose a generic energy model to evaluate and compare the energy consumed by these new Cloud architectures. We consider a scenario with V active VMs requested by a set of U end users. Our goal is to provide a generic model in order to estimate the energy consumption of each aforementioned cloud-related infrastructures for a given time period T when the allocated VMs are running. We do not take into consideration the differences among these architectures in terms of Quality-of-Service (i.e. latency). Besides, only the energy consumption of the infrastructure itself is estimated: it includes the telecommunication network between DCs and users but not the end users' devices.

As Figure II.10 shows, our model divides energy consumption of an ICT equipment into static and dynamic parts. The static energy consumption is the energy consumption without considering any workload (i.e. resources are idle). The dynamic cost is calculated based on the current usage of Cloud resources by the active VMs. The equation numbers on Figure II.10 refers to the original paper.

To reflect the energy consumption of non-ICT equipment available in data centers, the Power Usage Effectiveness (PUE) is a well-known data center energy-efficiency indicator. It represents

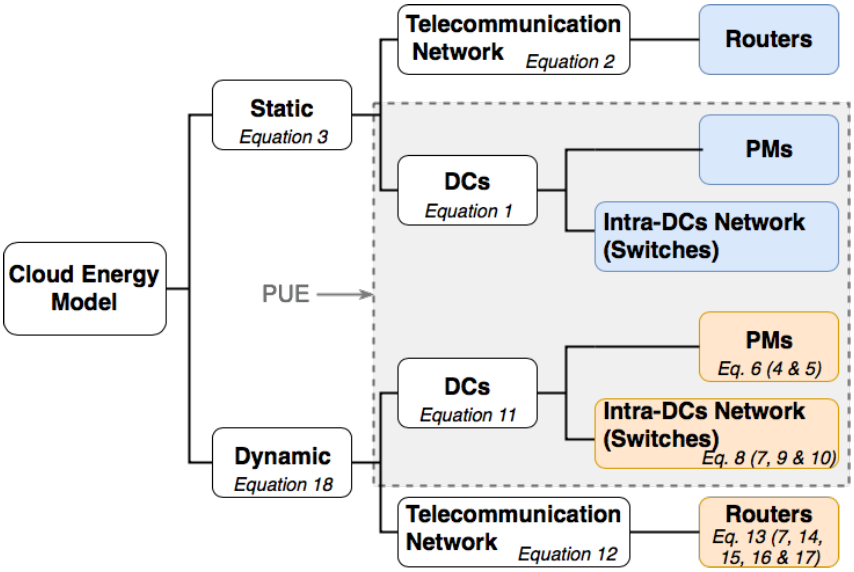


Figure II.10 – General view of the proposed energy model and the equations used in the following to express the different parts.

the ratio between the total facility and the ICT equipment energy consumption [ISO16]. In other words, the overall energy consumption of a data center can be estimated by multiplying the energy consumption of its ICT equipment and its PUE value.

As we consider the entire consumption of the network between data centers and end users, its energy consumption heavily dominates the total consumption. This is consistent with literature stating that telecommunication networks constitute the predominant part (37% in 2014) in the overall ICT consumption including end-user devices [HLL+14]. Future distributed Cloud architectures could reduce the need for network routers in keeping traffic as local as possible.

Moreover, the PUE greatly impacts the energy consumption of the architectures with medium and large-size data centers. Gains on the energy efficiency of ICT devices can be wiped out by a high PUE.

Although this model presents interesting insights on how to design the network architecture in highly distributed cloud infrastructures, it focuses on the cloud provider point of view.

II.C.2 VM models from user point of view

From a user point of view, relevant information concerns the power consumption and performance of her application. The work presented hereafter has been published in:

- "**An experiment-driven energy consumption model for virtual machine management systems**", Mar Callau-Zori, Lavinia Samoila, Anne-Cécile Orgerie and Guillaume Pierre, *Sustainable Computing: Informatics and Systems*, Elsevier, volume 18, pages 163-174, June 2018.
- "**How much does a VM cost? Energy-proportional Accounting in VM-based Environments**", Mascha Kurpicz, Anne-Cécile Orgerie and Anita Sobe, *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, Heraklion, Greece, pages 651-658, February 2016.
- "**Energy-proportional Profiling and Accounting in Heterogeneous Virtualized Environments**", Mascha Kurpicz, Anne-Cécile Orgerie, Anita Sobe and Pascal Felber, *Sustainable Computing: Informatics and Systems*, Elsevier, volume 18, pages 175-185, June 2018.

Figure II.11 presents a metric relative to power efficiency for a transactional web application.

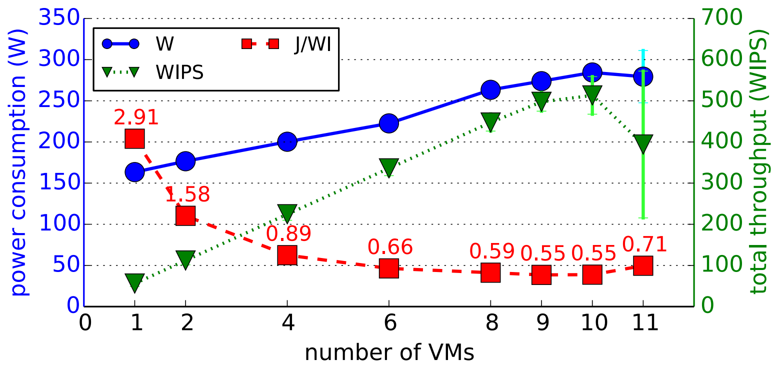


Figure II.11 – Power consumption and throughput when varying the number of VMs in the host.

The experiments are conducted on the Orion cluster of Grid'5000 [BCAC+13]. These servers embed two 6-cores Xeon E5-2630@2.3GHz processors (12 cores in total), 32 GB of RAM, 10 Gbps Ethernet, and a Nvidia Tesla M2075 GPU accelerator (not used in this case). We use Open-Stack [Ope] to dynamically create VMs with 4096 MB memory and 4 virtual CPUs. As the hosts considered in this experimental study comprise 12 cores each, without overcommit, they could only run three of such VMs each. We use TPC-W, a transactional web benchmark which simulates the activities of a business oriented transactional web server [TPC00]. TPC-W measures the throughput in number of WIPS (Web Interactions Per Second) that the server manages to sustain. This metric is employed to characterize the application performance.

The experiment shown in Figure II.11 evaluates how many VMs a single host can manage in an energy-efficient way. The scenario consists in increasing the number of VMs put on a single host. The X-axis shows the number of VMs in the host, whereas the Y-axes present three metrics: the power consumption in Watts of the entire server (blue circles, left side), the total throughput over all VMs in WIPS (green triangles, right side), and the power/performance ratio in Joules/WI (Joules over number of web interactions, red squares).

If focusing only on throughput, the most efficient configuration is the one ensuring linear scalability: so 8 VMs per host at maximum. However, maintaining a maximum of 8 VMs per host is not the most power-efficient case. This kind of power-efficiency metric, customized for a specific usage, targets user deploying a given application. However, it does not provide a generic framework for any kind of virtual application, and it practically depends on the good will of the Cloud provider to disclose the power consumption of its servers.

With colleagues from the Université de Neuchâtel (Switzerland), we addressed this issue by proposing EPAVE: a model for *E*nergy-*P*roportional *A*ccounting in *V*M-based *E*nvironmens. The idea consists to provide a fair and predictable model to attribute the overall energy costs per virtual machine (VM) in heterogeneous environments. EPAVE provides a full-cost model that does not account only for the dynamic energy consumption of a given VM, but also includes the proportional static energy cost of using a Cloud infrastructure.

If we consider the pay-as-you-go model as a basis, a VM would cost according to its size (i.e., resources reserved) and according to the time used. The same idea is followed by EPAVE, but we consider both static and dynamic energy as a basis of costs. Dynamic power consumption mainly depends on the resources which are used: computing, storage, networking resources. In the case of virtual environments, the hardware resources may be shared among different users and different virtual machines, if they run on the same host. In this context, a power-aware model needs to estimate the relative utilization per user to attribute the dynamic costs of the physical resources to a particular VM.

The main challenge lies in the division of the static costs among the users in a fair and predictable way, considering the utilization of the resources per VM. We showed that a simplistic model is not enough for distributing the costs among a number of VMs, as the static costs attributed to each VM would be highly dependent on the utilization of the same server (i.e. number of VMs). To ensure fairness among the users and predictability, our energy proportional accounting model is independent from the Cloud provider's VM management (not in control of the users): a given VM size executing a given application will get the same static cost from the EPAVE model even if executed at different dates on different servers.

As for dynamic costs, they can vary significantly from one server architecture to a different one. Performance and energy consumption heterogeneity among the servers is inherent to Cloud data centers. Typically, 3 to 5 server generations, with a few hardware configurations per generation, are hosted at the same time on a data center [DK13]; and this hardware heterogeneity leads to an important variability in terms of energy consumption as shown in Section II.B.1.

To show the applicability of our metric, we performed experiments using real-world applications on a taurus server of Grid'5000 [BCAC+13]. We installed Hadoop Yarn [Had] on each of the nodes and ran *sort* and *wordcount* from the HiBench [HHD+10] benchmark suite. We run the workloads within a VM to be able to limit the number of cores they use in total. We started the VM once with only a single core, and once with all the 12 cores available. As shown in Figure II.12, the static costs for using only a single core are smaller. However, because the single core is used for a longer time span, the dynamic costs are much higher leading to higher total costs than if all cores are used and reserved.

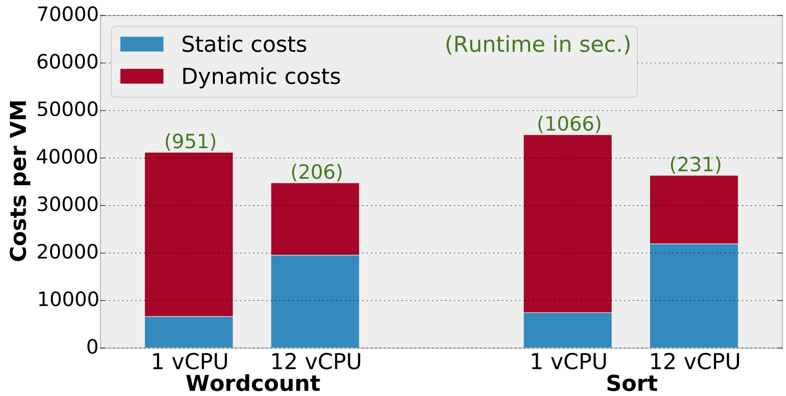


Figure II.12 – Costs of two parallel workloads with a VM of one vCPU and twelve vCPUs

EPAVE encourages users to dimension adequately their VMs. Indeed, if a user is asking for a 4 vCPUs VM, but uses only 2 vCPUs, the two unused vCPUs will still be taken into account into the static costs – although their dynamic costs will be zero, and even if the Cloud provider is applying over-commitment of resources. The user is also encouraged to be energy-efficient on its utilization of the resources. Indeed, the dynamic costs are directly measured from the hardware, so all energy saving mechanisms employed by the user (e.g., energy-aware software) will be directly translated into a reduction of the dynamic costs of the VM. We assume here that the energy costs of a VM have somehow repercussions for the user (like a bonus-malus system, or monetary costs for VMs taking into account the energy). While EPAVE is suitable for VMs inside a data center, it does not take into account data center locality and in particular, the network distance between the user and the data center.

II.C.3 End-to-end IoT-oriented models

Internet of Things (IoT) is bringing an increasing number of connected devices that have a direct impact on the growth of data and energy-hungry services. These services are relying on Cloud infrastructures for storage and computing capabilities, transforming their architecture into a distributed one based on edge facilities provided by Internet Service Providers. The work presented hereafter has been done in collaboration with Rutgers University (USA) and published in:

"End-to-end Energy Models for Edge Cloud-based IoT Platforms: Application to Data Stream Analysis in IoT", Yunbo Li, Anne-Cécile Orgerie, Ivan Rodero, Betsegaw Lemma Amersho, Manish Parashar and Jean-Marc Menaud, *Future Generation Computer Systems (FGCS)*, Elsevier, volume 87, pages 667-678, October 2018.

An IoT device does not consume a lot of power by itself, typically from few milliWatts to few Watts [SGSB+15, WNP11]. Yet, the increasing number of devices produces a scale effect and causes also a non negligible impact on Cloud infrastructures that provide the computing power required by IoT devices to offer services [AFGM+15]. To cope with the traffic increase caused by IoT devices, Cloud computing infrastructures start to explore the newly proposed distributed architectures, and in particular edge Cloud architectures where small data centers are located at the edge of the Cloud, typically in Internet Service Providers' (ISP) edge infrastructures.

While the current state of the art offers numerous studies on energy models for IoT devices [RS16, KL16] and Cloud infrastructures [JHA+16], to the best of our knowledge, we are the only ones to tackle the overall picture, with colleagues from Rutgers University. It is hard to estimate the energy consumption induced by the increase of IoT devices on Cloud infrastructures for instance. The issue resides in having an end-to-end energy estimation of all the involved devices and infrastructures, including network devices from ISP and Cloud servers. Such results could also serve to identify which part consumes the most, and should then attract the energy-efficient efforts.

The architecture of an IoT service is composed of several elements: the IoT devices themselves, the collecting point gathering the data from the IoT devices, the Cloud infrastructure used to process and to store the data and the network that link the collecting point and the Cloud. For the sake of clarity, we divide these components into three parts as depicted on Figure II.13:

- the IoT part comprising the IoT devices and the collecting point;
- the networking part comprising several switches and routers, their number depends on the Cloud architecture (centralized or edge);
- the Cloud part including the data center resources employed by the IoT service.

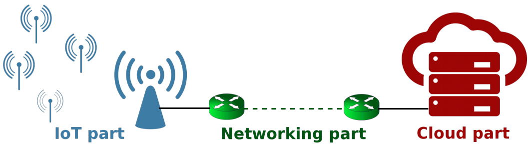


Figure II.13 – Three main infrastructure parts of an IoT service deployment.

Table II.1 – Estimation of the power cost per 360p stream for each part (using simulations for the IoT and network parts and real measurements for the Cloud part)

Scenario	IoT	Network	Cloud
Edge Cloud	10.96 Watts	0.07 Watts	32.3 Watts
Core Cloud	10.96	0.11 Watts	22.8 Watts

To evaluate our end-to-end model, the explored scenario is a camera-based monitoring service like a road traffic analyzer. Multiple cameras send data flows to the Cloud that processes them in order to detect objects on the road. The cloud itself can either be an edge cloud located near the cameras, or a core cloud located in a data center further away.

Table II.1 reports the computed power cost per stream for a 360p video for each part as defined in Figure II.13 depending on the use-case: edge Cloud or core Cloud. From this estimation, we can see that the predominant factor is the Cloud consumption (computing resources) in both cases: edge and core Clouds. But, in the case of the edge Cloud, it represents three quarters of the overall cost, while it represents two thirds for the core Cloud case. In both cases, the networking part is negligible, although routers are the most consuming devices per unit. Yet, if they are suitably loaded, their energy efficiency is high due to their large capacities. Finally, the IoT part, that includes the IoT device and the access point, accounts for one quarter of the overall cost for the edge Cloud case and one third for the core Cloud. These estimations advocate for a better energy efficiency of Cloud infrastructures.

In this study, along with the energy consumption, we also take into account several application-oriented performance metrics: the accuracy of the application (probability to detect an object with a given number of cameras using a given datarate) and the application delay in processing data streams (from the image capture to the object detection answer). Yet, for the energy side, we only rely on energy consumption without taking into account the electricity provenance that can be different between the core cloud and the IoT device. Furthermore, from a user point of view, providing raw energy values makes sense for assigning a value to the hidden energy consumption induced by her IoT device, but it gives no idea of how this device compares to others delivering the same service.

While some IoT devices produce a lot of data, like smart vehicles and cameras for instance, many others generate only a small amount of data, like smart meters or smart sensors. However, the scale matters here: many small devices can end up producing big data volumes. As an example, according to a report published by Sandvine in October 2018, the Google Nest Thermostat is the most significant IoT device in terms of worldwide connections: it represents 0.16% of all connections, ranging 55th on the list of connections [San18]. As a comparison, the voice assistants Alexa and Siri are respectively 97th and 102nd with 0.05% of all connections [San18]. This example highlights the growing importance of low-bandwidth IoT devices on Internet infrastructures, and consequently on their energy consumption. Our next step towards a comprehensive characterization of the global IoT energy footprint consisted in analyzing these low-bandwidth applications that periodically send few data to cloud servers. In this study, based on a smart sensor use-case, we show that for a given sensor, its larger energy consumption is on the sensor part, unlike the camera case described above. This work, not detailed in this manuscript, has been published in:

"**Estimating the end-to-end energy consumption of low-bandwidth IoT applications for WiFi devices**", Loic Guegan and Anne-Cécile Orgerie, *CloudCom: IEEE International Conference on Cloud Computing Technology and Science*, Sydney, Australia, December 2019.

II.C.4 CO_2 costs and ecolabels

The energy consumption of Cloud's data center causes greenhouse gas (GHG) emissions. This consequence is mainly determined by the amount and sources of consumed energy [BJKT16]. Among GHG, carbon dioxide (CO_2) is the major one in quantity produced by human activities. Consequently, carbon taxes have been proposed in order to reduce CO_2 emissions and their negative effects on environment [Nor12]. From an operational point of view, a carbon tax requires a monitoring and accounting infrastructure in order to fairly distribute CO_2 costs among the Cloud users. Even outside a carbon tax system, such an infrastructure can provide useful information to users about their real CO_2 emissions based on their utilization of the Cloud system, and therefore, it can raise their environmental awareness and incite them to adopt more sustainable practices. The work presented hereafter has been published in:

"**A CO2 emissions accounting framework with market-based incentives for Cloud infrastructures**", David Margery, David Guyon, Anne-Cécile Orgerie, Christine Morin, Gareth Francis, Charaka Palansuriya and Kostas Kavoussanakis, *SMARTGREENS: International Conference on Smart Cities and Green ICT Systems*, Porto, Portugal, pages 299-304, April 2017.

"**GLENDA: Green Label towards Energy proportioNality for IaaS DAta centers**", David Guyon, Anne-Cécile Orgerie and Christine Morin, *E2DC: International Workshop on Energy Efficient Data Centres (e-Energy Workshop)*, Hong Kong, pages 302-308, May 2017.

To build a carbon tax system, it is required to precisely monitor the resource usage that can be attributed to each user (computing, storage, communication), and to account for the resource cost induced by the user's utilization, like the data center air conditioning cost for instance. While the live monitoring issue has already been addressed in literature [WCP+15], the accounting issue has received little attention.

The accounting problem consists in splitting the indirect costs between the Cloud users (such as air conditioning), and forecasting the direct costs for each user. Indeed, Cloud computing is using a pay-as-you-go model where users buy computing, storage and network resources in the form of virtual machines (VM). Cloud providers exhibit prices per virtual machine type, depending on the amount of virtual resources included in the virtual machine. Such a model involves an *a priori* cost which is known by the user upon purchase as opposed to an *a posteriori* cost based on a precise monitoring of the resources really used and thus, provided to the user at the end of its Cloud resources utilization. Such an accounting model has to be flexible enough for the Cloud providers to be attractive, and it should provide to the users a predictable cost. From an external third-party organization, the carbon tax accounting system needs to be certified: for a given period of time, all the carbon emissions of the data center must be equal to the overall carbon emissions charged to the users.

Similarly to what we did for EPAVE in Section II.C.2, in this work, done in collaboration with the University of Edinburgh (UK), we propose a CO_2 emissions accounting model giving flexibility to the Cloud providers, predictability to the users and allocating all the carbon costs to the users. We go a step further in the applicability of the proposed solution by designing a framework architecture and ideas on how to practically implement it. In particular, we argue that instead of trying to keep the difference between predicted and real CO_2 emissions as low as possible at any time, an effective framework could consider this difference as a flexible capital to support an economical approach for users' energy-awareness.

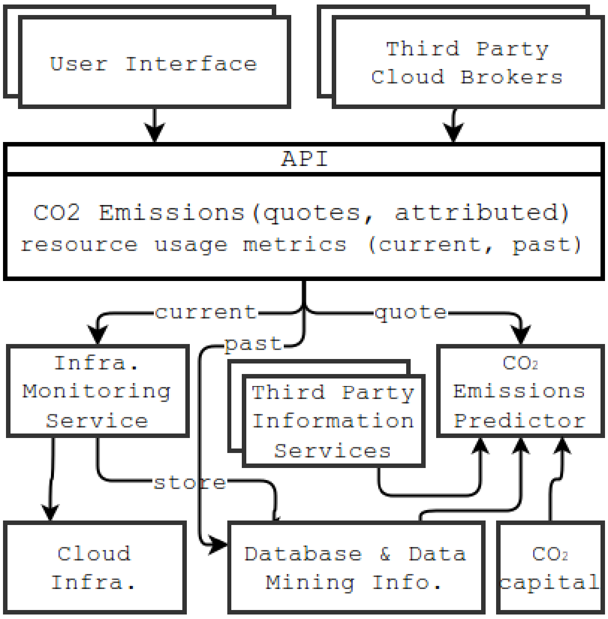


Figure II.14 – High level architecture of a CO_2 emissions accounting framework

Figure II.14 presents the high level architecture for enabling a provider to attribute CO_2 emissions to end-users. This system allows users to access information about resource usage (past and present), CO_2 emissions (estimated and attributed) for the VMs they run, and to quotes for CO_2 emissions that will be attributed to their future usage. Moreover, external services named Third Party Cloud Brokers can select platforms emitting the smallest amount of carbon between several Cloud providers to execute an application.

In order to lower the CO_2 impact, data centers managers have two options: increasing the share of renewable sources in their electrical mix, or increasing their overall energy efficiency. Yet, it is unclear which solution reduces the most the data center's impact. In our inquiry into easy-to-understand energy-related metrics, we proposed our own ecolabel, named GLENDA: Green Label towards Energy proportioNality for IaaS DAta centers. It assesses the energy-proportionality and green energy usage of Cloud's data centers. It combines two well-known data center metrics, namely the Power Usage Effectiveness (PUE) and the Green Energy Coefficient (GEC), the latter indicating the ratio of energy consumed from renewable sources.

Figure II.15 shows how we use GLENDA to compare various energy-aware approaches on real utilization and energy consumption traces from the Lyon site of Grid'5000:

- baseline: typical cloud management with current hardware and no power-saving technique.
- vary-on/vary-off (VO/VO): when a server is not used, it is powered down.
- power-proportional (PP): the power consumption of the servers is considered to be proportional to the utilization ratio. Thus, we consider that servers only have a dynamic power consumption that is reaching the maximal power consumption when the server is fully used, and that is null when the server is idle.
- max power (PP with P_{max}): this scenario expresses an infrastructure which is fully used at all time; whenever a server is utilized, its power consumption equals to the maximum server power consumption, and when unused, its consumption is null.

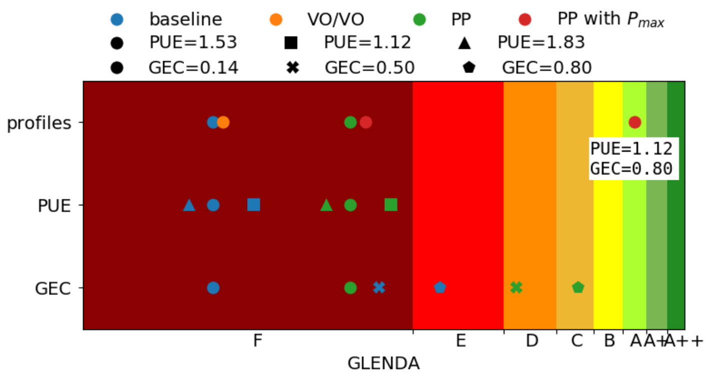


Figure II.15 – Utilization of GLENDA to compare baseline management, vary-on/vary-off management, power-proportional servers, and energy-efficient servers

The four markers at the top of the figure are the average GLENDA for the baseline, VO/VO, PP and max power scenarios. The second line shows the six markers for the baseline (in blue) and PP (in green) scenarios with three different values of PUE The bottom line exposes the average value of GLENDA for the same two scenarios but with three different values of GEC.

Figure II.15 shows that higher values of GLENDA are given whenever an infrastructure has power-proportional servers, its facility power consumption is near the total power consumption and when it consumes its energy from non-fossil sources. Such a green label for data center could on one side, spur Cloud providers to greater efforts for operating greener and more energy-proportional data centers, and on the other side, it could help energy-aware users to choose between different Cloud providers.

In this section, we explored comprehensive energy metrics for cloud providers and users. Although we looked at the applicability of our metrics in the context of public cloud providers, to the best of our knowledge, none of them currently offer such monitoring services for clients. The only accessible energy-related metric concerns the PUE that some providers display (e.g. Google[6],

[6]Google data center PUE: `https://www.google.com/about/datacenters/efficiency/internal/`

Facebook[7], Iliad[8]). But, PUE alone does not help users to evaluate the energy consumption of their own application, or to compare the CO_2 impact of two service deployment solutions.

II.D Towards comprehensive simulation tools

In most of the work presented hitherto, we rely on the resources of the Grid'5000 platform to carry out life-size experiments and to validate our energy models and metrics on heterogeneous resources equipped with wattmeters. However, it is usually difficult and expensive to access large-scale distributed infrastructures. Simulating distributed infrastructures is thus essential for validating new energy-efficient solutions. Indeed, simulation gives access to possibilities that experimental platforms often do not offer, or not enough, in terms of scalability, geographical distribution or heterogeneity.

In the design of energy-efficient algorithms, it is necessary to know the energy consumption of resources. However, instrumenting distributed infrastructures remains expensive in measurement equipment, in deployment time and in software development to give access to the data. Simulation can offer reproducibility guarantees and allows reliable and fair comparisons between different algorithms.

Like for the metrics presented in the previous section (Section II.C), numerous cloud simulator exists, but none provides comprehensive energy estimations. In this section, I showcase my contributions to simulation frameworks towards this end. The journey started with the network simulator ns-3 (Section II.D.1), it continued with the distributed computer systems simulator SimGrid (Section II.D.2), and it made a recent visit into co-simulation frameworks with SimGrid and OpenModelica (Section II.D.3).

II.D.1 Network simulator

My first contribution to energy-aware simulation tools started during my PhD thesis with the Ecofen framework (End-to-end energy Cost mOdel and simulator For Evaluating power consumption in large-scale Networks) proposed in 2011 [OLGLLP11]. In 2013, I refined and re-designed it for ns-3 [ns3], a discrete-event simulator for Internet systems, targeted primarily for research and educational use. Since then, Ecofen has been used by several research teams for studying various networking infrastructures. The work presented hereafter has been done in collaboration with colleagues from Inria Lyon and University of Nice, and published in:

"**Simulation toolbox for studying energy scenarios in wired networks**", Anne-Cécile Orgerie, Betsegaw Lemma Amersho, Timothée Haudebourg, Martin Quinson, Myriana Rifai, Dino Lopez Pacheco, and Laurent Lefèvre, *CNSM: International Conference on Network and Service Management*, Tokyo, Japan, pages 1-5, November 2017.

The main goal of the Ecofen toolbox is to provide a simulating environment for large-scale wired networks, where users can obtain the energy consumption of their new protocols, algorithms and frameworks involving different types of technologies and equipment. It is implemented as an ns-3 module plugged on the network devices and ports' abstractions provided by ns-3.

Ecofen is endowed with several energy models, and several representative network devices are pre-defined using energy consumption values found in the literature. The energy models for network ports offer several parameters: energy consumption per processed packet, energy consumption per processed byte, idle power consumption, sleeping power consumption, and energy consumption to switch on and off a port. The energy models for routers add parameters to take into account the

[7] Facebook Prineville data center PUE: `https://www.facebook.com/PrinevilleDataCenter/app/399244020173259/`

[8] Iliad DC3 Vitry data center PUE: `https://pue.online.net/fr`

power consumption for chassis and linecards while being idle, sleeping or switching on and off. Ecofen also comprises energy efficient levers such as rate adaptation mechanisms (ALR), sleeping mechanisms (LPI), coalescing approaches and switching on and off functions. Figure II.16 provides an example of the energy consumption model for switching off and on a network port with Ecofen.

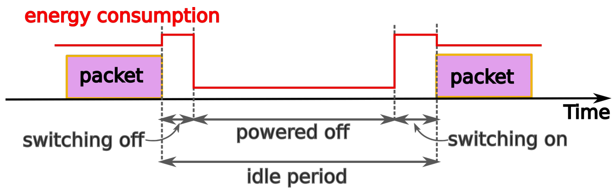


Figure II.16 – Example of the energy consumption model of Ecofen for switching off and on a network port.

We validated the Ecofen simulation results against real measurements performed on network devices from the literature by lack of an adequate measurement infrastructure. This validation gave satisfactory results presented in the original paper. This tool allow us to produce the simulation presented in Section II.B.2. It has also been used by colleagues at the University of Nice to study the energy consumption in the core and access networks using Software Defined Networking protocols (and we should remember in a near future to think about publishing these results).

Ecofen has been my first step in the simulation world, but it was limited, as ns-3 only provides abstractions for network resources. To simulate entire Cloud infrastructure, it lacks at least of models for computing resources.

II.D.2 Cloud simulator

I pursued my goal of offering simulation tools embedding accurate energy models for Cloud infrastructure-scale systems by exploring other simulation tools. The work presented hereafter has been done in the context of the Hac Specis project, and published in:

"**Predicting the Energy-Consumption of MPI Applications at Scale Using Only a Single Node**", Franz C. Heinrich, Tom Cornebize, Augustin Degomme, Arnaud Legrand, Alexandra Carpen-Amarie, Sascha Hunold, Anne-Cécile Orgerie and Martin Quinson, *IEEE Cluster Conference*, Hawaii, USA, pages 92-102, September 2017.

"**A Large-Scale Wired Network Energy Model for Flow-Level Simulations**", Loic Guegan, Betsegaw Amersho, Anne-Cécile Orgerie and Martin Quinson, *AINA: International Conference on Advanced Information Networking and Applications*, Matsue, Japan, pages 1047-1058, March 2019.

Following the measurement campaign presented in Section II.B.1 on Grid'5000 servers to analyze the relation between power consumption and CPU utilization, we proposed a model of this relation and we implemented it within SimGrid [Sima], a simulation framework for distributed applications coming either from HPC or Cloud computing. It is indeed essential for both communities to ground their simulation tools on sound server models.

FigureII.17 compares SimGrid simulations and real executions for the three applications presented earlier: NAS-EP, NAS-LU and HPL (see Section II.B.1). In all cases, we manage to systematically predict both performance and energy consumption within a few percents.

We also recently added energy models for wired networks within SimGrid in order to get, from the same simulation tool, the energy consumption of both computing and networking resources. This implementation was validated against the Ecofen module of ns-3 presented in the previous

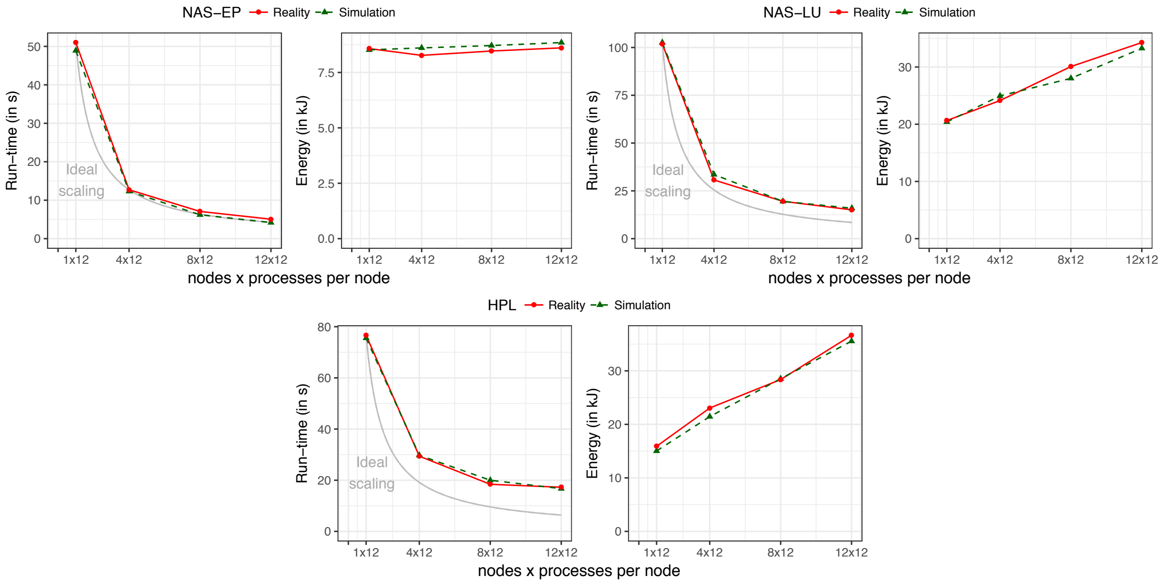
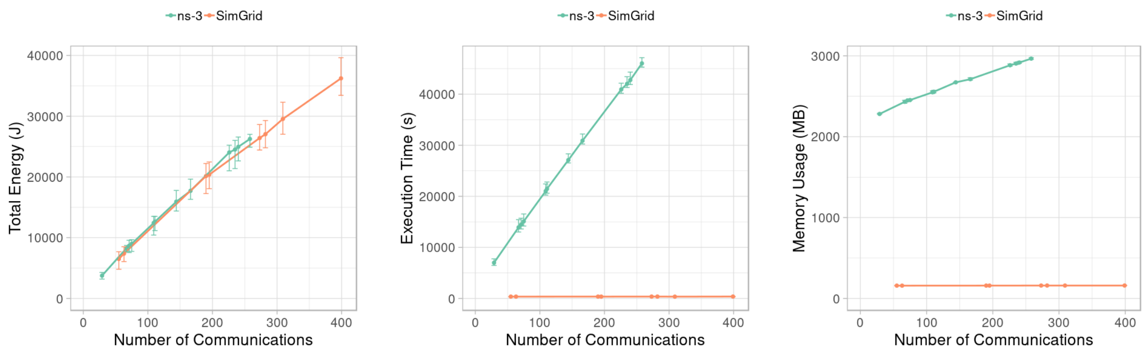


Figure II.17 – Validating simulation results for NAS-EP, NAS-LU, and HPL, on up to 12 nodes with 12 processes per node.

section (Section II.D.1). Notably, it required to switch from a packet-level model (ns-3 model) to a flow-based model (SimGrid model) that does not simulate each packet, but rather data flows.

To evaluate the scalability and accuracy of our approach, we simulate a datacenter' network with its classical three-tier architecture.

Figure II.18 show the validation results of SimGrid against ns-3 on a scenario simulating a data center network comprising more than 1,500 servers and random communication among them. Figure II.18(a) only displays the dynamic energy consumption of the network devices, as it constitutes the difficult part to simulate, since the fixed or static power consumption uses the same model (i.e. a constant value) in both simulators. Figures II.18(b) and II.18(c) respectively present the execution time and the memory usage of each simulation for comparing the performances of the two simulators. We obtain a precision close to ns-3 with less than 4% relative error on the dynamic energy consumption, and with simulation runtime 120 times faster on flow-level simulators. This realistic use case highlights how SimGrid can now be employed by the scientific community on large-scale platforms to simulate the energy consumption of wired networks. The implementation is open-source and available on the SimGrid website [Sima]. Our complete validation experiments can be found here: `https://gitlab.inria.fr/lguegan/flowlvlwiredenergy`.



(a) Overall data center power profile (b) Simulations execution time (c) Simulation memory usage

Figure II.18 – Energy and scalability simulations results

II.D.3 Co-simulation framework

We went a step further in our quest towards comprehensive simulators for Cloud infrastructures when we targeted data centers and their cooling systems. The work presented hereafter has been done during the post-doc of Benjamin Camus within the COSMIC project[9], and published in:

"**Co-simulation of FMUs and Distributed Applications with SimGrid**", Benjamin Camus, Anne-Cécile Orgerie and Martin Quinson, *PADS: ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*, Roma, Italy, pages 145-156, May 2018.

In this work, we consider a data center and its chiller and we simulate their working relations. When the chiller demand (which depends on the heat dissipation induced by computations) becomes too high, a safety mechanism shuts down the power supply to lower the temperature and to preserve servers. We simulate the computing processes which cause and handle this mechanism. It requires to model both the computing load of the data center, and the physical processes of heat transfers. We use SimGrid to simulate the computing load and its power dissipation, through the models implemented in Section II.D.2. As SimGrid does not include thermal models, we use another simulator for this part, namely OpenModelica, an open-source Modelica-based modeling and simulation environment [OM].

Figure II.19 illustrates this scenario that implies:

1. coupling different modeling and simulation tools (OpenModelica and SimGrid),
2. which use different modeling paradigms (algebraic/differential/discrete equations and concurrent programs),
3. with discrete (distributed application execution) and continuous (the temperature evolution) dynamics in interaction (the distributed application changes the servers' heat dissipation, and the room temperature triggers power shutdown that kills the running programs).

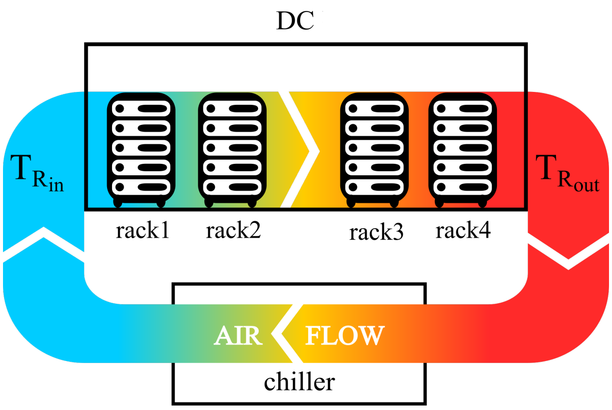


Figure II.19 – Simulated system: a data center with its computing resources and its chiller.

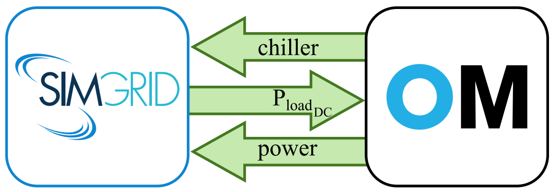


Figure II.20 – Data exchanges between SimGrid and OpenModelica (OM)

Figure II.20 shows the interactions between SimGrid and OpenModelica. For coupling the two simulators, we rely on the Functional Mock-up Interface (FMI) standard [BOÅ+12] of the Modelica Association that offers a unified framework and an API to control equation-based models of multi-physical systems (e.g. electrical, mechanical, thermal systems). This standard is supported by over 100 modeling and simulation tools[10].

Using FMI, a model which may be composed of a mixture of differential, algebraic and discrete-time equations, can be exported under a standard format as an FMU. This FMU is a black-box with

[9] Inria exploratory action on Coordinated Optimization of SMart grIds and Clouds (2016 - 2018) `http://people.irisa.fr/Anne-Cecile.Orgerie/COSMIC/`

[10] according to http://fmi-standard.org

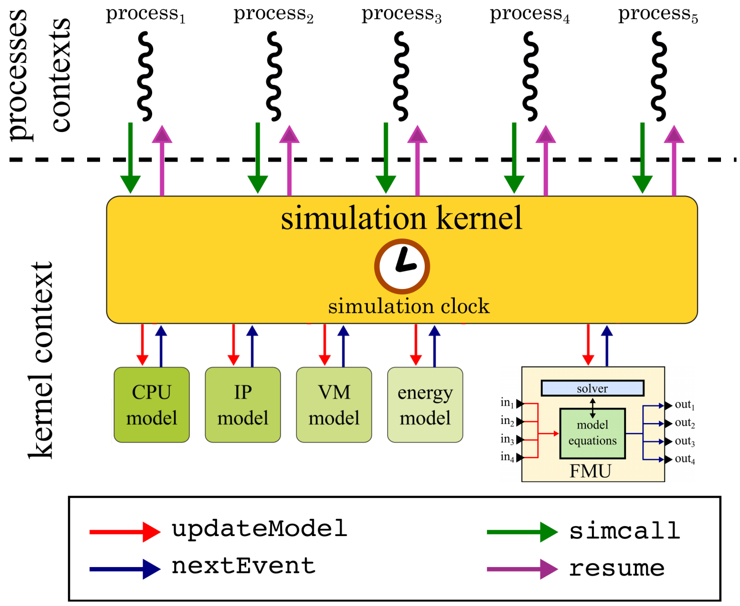


Figure II.21 – Simulation architecture of SimGrid.

input and output ports which correspond to the input and output variables of the model. Each FMU can then be controlled using a standardized API, regardless the simulation tool used to generate it. An FMU can also directly integrate a passive solver that can be controlled by any simulation environment importing this FMU.

The co-evolution of an FMU with its environment is based on the concept of communication points. These communication points, which have to be set by the environment of the FMU, correspond to points in the simulated time where (1) the FMU simulation must be stopped, and (2) exchanges of data can be performed between the FMUs and its environment. Between two communication points, an FMU evolves independently of its environment.

Our idea consists in using SimGrid as a cosimulation framework. Figure II.21 shows the proposed cosimulation architecture. During a typical SimGrid simulation, all user processes are conceptually executed in parallel. A simulation kernel, which has its own execution context, is in charge of:

1. managing the simulation state (e.g. the simulation clock),
2. coordinating the processes and models' executions, and
3. mediating interactions between user code and models.

The SimGrid models are based on the discrete event paradigm, where an internal event corresponds to the completion of an action. As illustrated on Figure II.21, our approach consists of importing the FMU into a dedicated model which is added in the SimGrid simulation kernel. The kernel can then control the FMU like any other model. Note that with this mechanism, several FMUs can be imported in SimGrid, each of them being associated with a dedicated model. All the FMUs can then interact separately with the distributed application processes.

The original paper exhibits a validation of our solution by demonstrating that our co-simulation of a data center's computing workload and its chiller gives similar results when compared to a monolithic simulation. Thanks to this work, SimGrid is now able to interact with FMI-compliant modeling and simulation tools. We are currently working on using this feature to co-simulate distributed Cloud infrastructures and electrical networks. Further details on this work will be provided in Section IV.D.3.

Our incremental approach towards comprehensive simulation tools for Cloud infrastructures should be considered in the long term as it requires numerous coding hours and experimental validations (real measurements vs. simulations matches can be harsh). Yet, it provides theoretically sound and experimentally assessed simulation models on top of which PhD students can confidently build their validation tests.

These contributions also produce a unique tool able to simulate entire data centers with servers, network devices, their respective power consumption, chillers, and soon electrical networks. Retrospectively, it seems worth the investment. Furthermore, the actual implementation of the models, observed during the measurements campaigns, and their validation contribute to the fine understanding of the physical phenomena at stake. I definitely gained a lot of comprehension about the energy consumption of Cloud systems in this process.

II.E Perspectives

This chapter synthesizes my steps towards understanding the energy consumption of distributed infrastructures. I chose to adopt an experimental approach, starting with measurements and experiments on real platforms. However, this would have been unfeasible without the involvement of numerous colleagues, mostly post-doctoral researchers (for this chapter of the manuscript) for these highly technical investigations. It would also have been unachievable without an adequate experimental testbed. Grid'5000 [BCAC+13] constitutes a unique and valuable platform which, I hope, will continue to favor many generations of post-docs and PhD students. This hope led me to invest time to serve as the chief scientist of the Rennes site since 2018. Indeed, since 2012, although I still find experimentation as an amazing adventure, I softly (but surely) drifted on the other side: supervising others' experiments rather than handling them myself.

Reproducible experimental approach. This concrete experimental approach can somehow be disappointing at first, especially after multiple runs of the same experiments, and still no consistent values or data beyond understanding. Yet, it provides invaluable practice to both (1) performing reproducible experiments, and (2) comprehending energy consumption in distributed infrastructures. Concerning the first point, reproducibility is a quite recent issue for computer science, but it deserves all of our attention, the credibility of our research is at stake. Since many parameters can influence the power consumption, power measurements constitute a nice usecase to try and test the robustness of an experimental methodology.

On the second point, these experimental insights allowed me, for instance, to help colleagues from Bordeaux and Sophia to analyze the energy profile of their HPC application. This contribution, not detailed in this manuscript, has been published in:

"**Energy Analysis of a Solver Stack for Frequency-Domain Electromagnetics**", Emmanuel Agullo, Luc Giraud, Stéphane Lanteri, Gilles Marait, Anne-Cécile Orgerie and Louis Poirel, *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, Pavia, Italy, pages 385-391, February 2019.

With the rapid evolution of ICT infrastructures, new computing paradigms appear, bringing original systems to monitor. My future work in this direction includes measuring the energy consumption of emerging ICT infrastructures, and devising models for the distinct elements of these systems. Concretely, I currently work with colleagues, on measuring the energy consumption of computing in the continuum [AZZ+17]. Towards this end, we started measuring the energy consumption of GPU architectures in the PhD thesis of Dorra Boughzala, co-advised with Laurent Lefèvre and Martin Quinson.

Fine energy modeling. The measurement starting point then led me to design energy models, with a focus on comprehensive models including indirect energy impacts. Models are required by Cloud users and providers at different scales (e.g. infrastructure-wide, server-oriented). My contributions concern different perimeters depending on the targeted utilization, for instance: CO_2 cost, IoT device's overall consumption, infrastructure sizing. I also contributed to energy models for data-intensive applications in the HPC domain. This work, not detailed in this manuscript, has been published in:

"**On the Energy Footprint of I/O Management in Exascale HPC Systems**", Matthieu Dorier, Orçun Yildiz, Shadi Ibrahim, Anne-Cécile Orgerie, and Gabriel Antoniu, *Future Generation Computer Systems*, Elsevier, volume 62, pages 17-28, September 2016.

Energy consumption modeling, in the area of distributed ICT infrastructures, is challenging due to the virtual nature of resources and the infrastructure sharing. This domain keeps many open issues of great interest for companies, like auditing the energy consumption of a given digital service for instance, such a service being split among multiple virtual machines spread across several data centers.

Sound comprehensive simulation tools. From measurements, through models and to simulation tools, the journey is not linear, although it has to be presented in a linear way in this manuscript. Simulators mainly deal with building the right tools to answer scientific questions. In our context of distributed Cloud infrastructures, it involves many components and models, and even more potential applications and usecases. As an example, I also contributed to the simulation of the energy consumption of I/O intensive scientific workflows with the colleagues developing Wrench, a workflow management system simulation workbench built on top of SimGrid. This work, not detailed in this manuscript, has been published in:

"**Accurately Simulating Energy Consumption of I/O-intensive Scientific Workflows**", Rafael Ferreira da Silva, Anne-Cécile Orgerie, Henri Casanova, Ryan Tanaka, Ewa Deelman and Frédéric Suter, *ICCS: International Conference on Computational Science*, Faro, Portugal, pages 138-152, June 2019.

SimGrid community has now access to energy consumption for servers and wired network devices. It is also possible to use ns-3 models within SimGrid directly, since the ad-hoc coupling done few years ago which allows to simulate wired links within ns-3 inside a SimGrid simulation. Yet, ns-3 being much slower than SimGrid (as shown in Section II.D.2), this solution does not suit to simulating large-scale topologies with numerous large communications. Following on from these implementation efforts, I contribute to an ongoing work on integrating WiFi communication models within SimGrid. Following the flow-based modeling philosophy of SimGrid, this implementation should enable large-scale and fast simulation of numerous WiFi devices, thus making the myriads of IoT and Fog objects accessible to thorough observation under our microscope. This work is part of the PhD thesis of Loic Guegan co-advised with Martin Quinson. Accurate simulation tools can indeed provide meaningful insights on the functioning of large-scale systems.

As more and more researchers get concerned by energy consumption, accurate simulation tools are required. In the complex case of distributed computing infrastructures, co-simulation is a promising solution, getting the best of both worlds. Comprehensive simulators can then serve to realize application-oriented what-if scenarios in order to design new infrastructures or to improve existing ones. For instance, such tools could determine the least energy-consuming deployment of devices for smart infrastructures, such as buildings or factories, monitored by numerous ICT devices spread across the studied infrastructure.

Mathematicians are like managers; they want improvement without change.
Edsger Dijkstra

III
Improving the energy efficiency of distributed infrastructures

III.A Introduction to energy efficiency

My second research axis was focused on improving the energy efficiency of distributed infrastructures. Energy efficiency involves performance metrics or a measure of useful output per energy unit. Indeed, energy can not be the only criterion to take into account, at the risk of losing unsatisfied users. It is therefore necessary to put in place energy-efficient policies in respect with the desired quality of service, ensuring the satisfaction of both users and resource providers.

Various methods have been proposed to increase energy-efficiency, at both software and hardware levels: variation of the frequency of the processor as a function of the load (i.e. Dynamic Voltage Frequency Scaling), extinction of unused cores of computation, consolidation of the load on a limited number of servers to shut down unused ones (i.e. shutdown approach), etc. These techniques presents incompatibilities: some cannot be combined to be used simultaneously on a given server (e.g. impossible to change the frequency of a turned-off processor).

Cloud infrastructures comprise numerous hardware and software components. Employed judiciously, energy-efficient techniques can allow consequent energy savings. In particular, the shutdown approach has the capacity to consequently lower the idle power consumption of unoccupied servers. Indeed, it still represents numerous Watts, as shown in Figure III.1 (baseline taken from measurements presented in Section II.B.1). This high idle power consumption leads to the non-power proportionality of servers, and consequently to their poor energy efficiency during low usage phases. The shutdown approach targets these inefficient Watts and aims at switching off idle servers.

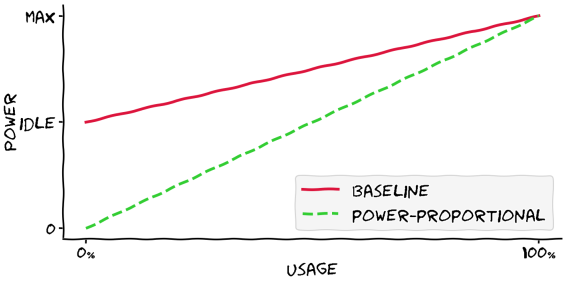


Figure III.1 – Non-power proportionality of current servers

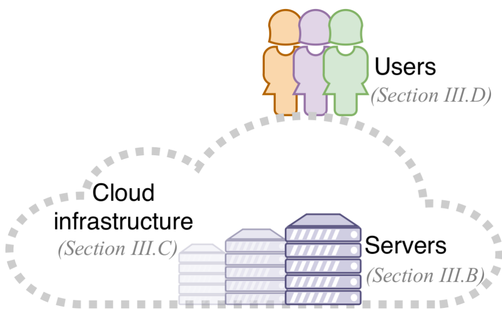


Figure III.2 – Schematic view of the energy-efficient levers location for the Cloud systems

On top of the servers, the architecture of a given computing system can also be intrinsically inefficient in terms of energy consumption. Stepping aside and rethinking this architecture could help to reduce its consumption. For instance, it seems interesting to explore in detail distributed architectures and to revisit the existing mostly-centralized data center management policies to improve the overall energy efficiency. In particular, network resources are often neglected in the literature although they may be a major consumption item.

Energy-efficient techniques or architectures may impact the system's functioning and consequently, its quality of service. To keep users' satisfaction at a high level, they could be involved in the management policies to apply energy-efficient techniques. For instance, an Internet service could ask one of its user to shift their utilization slightly over time if it allows to utilize one server less.

As in Section II.B, I adopt in this chapter a bottom-up plan to describe my contributions, illustrated on Figure III.2. Starting with servers, I present techniques to fight against the non-power-proportionality of computing resources, and in particular the shutdown approach (Section III.B). Then, I explore new decentralized Cloud infrastructures and their associated cloud stack (Section III.C). Thirdly, I seek for ways to involve users in the energy-efficiency quest (Section III.D). Finally, Section III.E presents perspectives on this work.

III.B Fighting the non-power-proportionality of computing resources

Despite the associated financial cost for their operators, a large number of data centers spend the majority of their time at utilization levels varying from 10% to 50% [BCH13], which stems from infrastructure over-provisioning and allocated resources that are not fully utilized. As a result, not only resources, but also energy is wasted since a server can consume over half of its peak power consumption when idle as shown on Figure II.3. As fully energy-proportional servers do not exist yet, increasing energy efficiency relies on keeping servers utilization at a high level and switching off unused servers. This energy-efficient technique was in particular explored during the PhD thesis of Issam Raïs (October 2015 - September 2018) that I co-advised with colleagues from Lyon (Laurent Lefèvre and Anne Benoit) within the context of the ELCI project[1].

III.B.1 Energy costs and gains of switching off servers

The work presented hereafter has been published in:

"**Impact of Shutdown Techniques for Energy-Efficient Cloud Data Centers**", Issam Raïs, Anne-Cécile Orgerie and Martin Quinson, *ICA3PP: International Conference on Algorithms and Architectures for Parallel Processing*, Granada, Spain, pages 203-210, December 2016.

"**Quantifying the Impact of Shutdown Techniques for Energy-Efficient Data Centers**", Issam Raïs, Anne-Cécile Orgerie, Martin Quinson and Laurent Lefèvre, *Concurrency and Computation: Practice and Experience*, Wiley, volume 30, issue 17, September 2018.

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. It exists 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 III.3 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, followed by an *Off* section, and then an *Off* to *On* sequence. The

[1] ELCI: PIA project on environment for computation-intensive applications (2014 - 2017) `http://elciproject.unblog.fr`

bottom graph represents the same machine in Idle state for the same time period. 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.

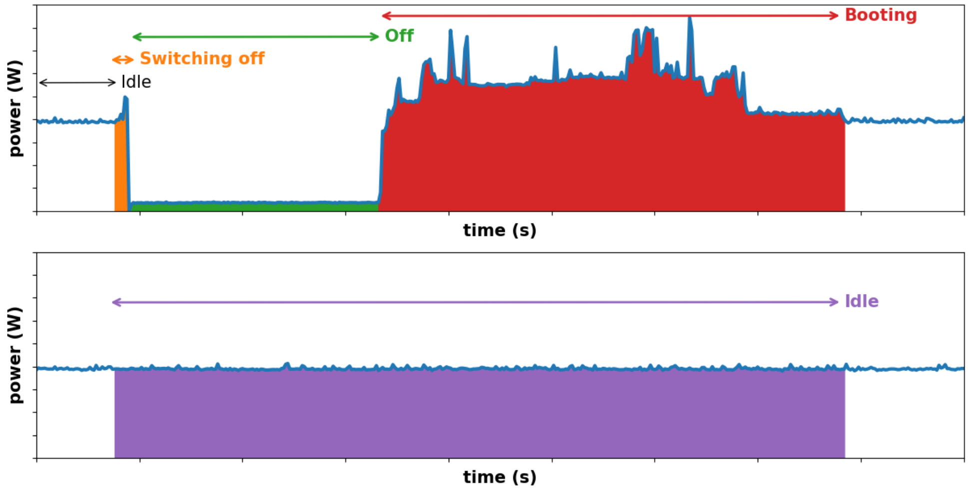


Figure III.3 – Time threshold to decide whether to switch off or not

Following this principle, T_s is defined as:

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

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 servers. Indeed, as future is not known in the general case, predictions are required to determine for a given idle data center server if it will stay idle for more than T_s or not.

We computed these values for 3 different Grid'5000 clusters (Orion, Taurus, Paravance) [BCAC+13] while performing switching off and on operations. The servers are running a standard Debian Jessie (Debian GNU/Linux 8.0 for x64 architectures). For these servers, inactivity periods between 2 and 4 minutes are sufficient to save energy by switching off. However, one major 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 for HDD [Sea12].

In order 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 bounds for energy consumption. Our evaluations rely on replaying utilization traces from real data centers and using the three servers power profile of Grid'5000 that we measured. *Policy P1: knowing the future:* this policy considers that the future is completely known. Thus, dates and lengths of idle period are known for each server. *Policy P2: aggressive shutdown:* this 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 P1 because some idle periods may be lower than T_s. In such cases, switching off increases the energy consumption compared to staying idle. *Ideal PP:* 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 ($P_{off} \neq 0$).

Table III.1 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).

Table III.1 – 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 PP	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 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 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 [Sea12]. This clearly states that even aggressive shutdown policies have no impact on disk lifetime despite the common belief.

It is worth noticing that significant energy gains can be performed for both traces even though they present completely different use cases. Indeed, 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).

Although this study provides theoretical results with an *a posteriori* replay of workload traces, it exhibits the potential energy savings reachable through the use of shut down policies. Yet, to be feasible, we have to investigate the practical consequences of switching off and on equipment in data centers. It is important to notice that Grid'5000 already enforces a shutdown policy on its servers, and that these servers already perform reboots quite often (as soon as a user wants to deploy its own environment image on a server) without particular issues on their lifetime (e.g. currently the oldest cluster on Rennes site has been bought in January 2010 and 20 out of the 25 initial servers are still properly working).

III.B.2 Constraints in switching off hardware resources

Shutdown policies constitute an appealing approach able to dynamically adapt the resource set to the actual workload. However, multiple constraints have to be taken into account for such policies to be applied on real infrastructures: the time and energy cost of switching on and off, the power and energy consumption bounds caused by the electricity grid or the cooling system, and the availability of renewable energy. The work presented hereafter has been published in:

"**Shutdown policies with power capping for large scale computing systems**", Anne Benoit, Laurent Lefèvre, Anne-Cécile Orgerie and Issam Raïs, *Euro-Par: International European Conference on Parallel and Distributed Computing*, Santiago de Compostela, Spain, pages 134-146, August 2017.

"**Reducing the energy consumption of large scale computing systems through combined shutdown policies with multiple constraints**", Anne Benoit, Laurent Lefèvre, Anne-Cécile Orgerie and Issam Raïs, *International Journal of High Performance Computing Applications*, SAGE, volume 32, issue 1, pages 176-188, January 2018.

Datacenters gather servers, switches and cooling systems, and deals with an electrical provider to power its infrastructure. In practice, turning off too many nodes could cause the temperature to be lower than the optimal temperature bound, and the power used to be under the minimum power capping negotiated with the electrical provider. Likewise, if too many nodes are turned on, and if the energy consumed during shutdown and wake-up sequences is taken into account, limits fixed by the power provider can greatly be overcome and at the same time, could cause the temperature to raise drastically, creating hotspots. If such constraints are not taken into account, they can put into danger machines composing the operational computing facility.

To deal with these issues, we propose a framework that models server shutdown process under various constraints. It takes into account the impact of On→Off and Off→On sequences in terms of time, power and energy. It also takes into account idle and off states observed after such sequences, since they deeply impact the electrical usage of resources Our framework allows to combine constraints in order to help resource managers and providers to respect several constraints at the same time.

The proposed models of shutdown policies are the following:

- The *basic models* allow comparisons with several related works where turning on and off can be immediate and free of energy consumption.
- The *sequence-aware models* focus on the On→Off sequences when providers want to switch off several useless resources and to switch them on again when these resources are needed. These models deal with the availability of scheduling On→Off sequences during gaps and their potential energy benefits.
- The *electricity-aware models* deal with the electrical provision of the data center in order to avoid large-scale aggressive electrical demands (due to massive switch on of resources) and to respect power capping requirements.
- The *cooling-aware models* respect the constraints imposed by the cooling infrastructure of the data center. They follow the thermal constraints of the system by reducing the number of possible On→Off sequences.
- The *renewable-energy-aware models* support selective shutdown policies by considering the electricity provenance (from renewable energy or from fossil-based energy sources).

While it is often assumed that nodes can be turned off at no cost, we explore realistic scenarios where several constraints (power capping, electricity, thermal) may prevent from turning off a node at a given time. A possible usage of these models is illustrated in the original paper through a set of simulations on a real workload trace, showing the gain in energy that can be achieved given the constraints of the platform, and providing clear guidelines about when each server can be turned off. When shutdown policies are not applicable, other solutions have to be investigated to fight against the non-power proportionality of servers.

III.B.3 Alternatives to switching off

As an alternative to switching off non-power-proportional and energy-hungry servers, we explored solutions to take advantage of the produced heat. In a data center, great part of consumed energy is lost in exothermic emissions. For the safety of data centers, this lost energy is carried away with air or water cooling systems. ThermoElectric Generators (TEGs) aim to recover energy by converting wasted dissipated energy into usable electricity. The work presented hereafter has been published in:

"**An analysis of the feasibility of energy harvesting with thermoelectric generators on petascale and exascale systems**", Issam Raïs, Laurent Lefèvre, Anne Benoit, and Anne-Cécile Orgerie, *International Workshop on Optimization of Energy Efficient HPC & Distributed Systems (OPTIM)*, in conjunction with HPCS, Innsbruck, Austria, pages 808-813, July 2016.

"**Quantifying the Impact of Shutdown Techniques for Energy-Efficient Data Centers**", Issam Raïs, Anne-Cécile Orgerie, Martin Quinson and Laurent Lefèvre, *Concurrency and Computation: Practice and Experience*, Wiley, volume 30, issue 17, September 2018.

A thermo electrical material transforms a temperature difference into electricity. TEGs are composed by positively (p-type) and negatively (n-type) doped connected semiconductor couples. N-P couples are the charge carriers that can freely move through the metal. These carriers start to move under a temperature discrepancy, according to the N-P couple properties. The temperature difference creates an excitation of the doped charge carrier, thus inducing a movement of the charge carrier, creating an electric current. A larger temperature difference produces a larger electrical current, but this statement is highly coupled with the fact that a semiconductor is effective only on a range of temperature, making the TEGs operational only on a limited temperature difference [ST08].

We studied the potential gains in combining TEGs with servers at large scale. Current TEG that are suitable for this scenario have a low efficiency (at maximum in ideal conditions around 12%). Yet, our study show that TEGs could be profitable after approximately 3 years of usage under ideal conditions in a supercomputer context. This study puts in balance the saved electricity costs against the buying costs of TEGs. However, it does not include the cost of installation and maintenance on existing data centers. Beyond its cost, the installation of TEGs can be prohibitive for safety and complexity reasons. Although theoretically appealing, this solution stays far from being workable.

Another promising alternative to switching off techniques consists in increasing the heterogeneity of computing resources to better suit the demand. The ARM big.LITTLE processor is an example of such a promising solution in terms of energy-efficiency. It combines low-power processors with high-performance ones to offer an heterogeneous architecture closer to power proportionality than other processors even with dynamic frequency scaling [Jef12]. The idea consists in activating one kind of processor at a time: either the low-power ones during low workload or the powerful ones during high activity.

Following the same concept, we consider heterogeneous data centers with servers offering low, middle and high (i.e. regular) computing capabilities and their respective power consumption to mimic the potential configuration of future energy-proportional data centers. We evaluated their overall energy consumption combined with the energy policies and on the real traces used in Section III.B.1. As expected, switching off only the little or the medium components results in little energy savings. Yet, whenever the big unit can be switched off, consequent amounts of energy are saved, showing that switching off techniques would be useful for such envisioned architectures.

I also contributed to the study of other energy leverages at server level, such as parallelization with multi-threading, computation precision (i.e. int, float, double) and vectorized instructions (SSE, AVX, AVX2, AVX512). Details on this work can be found in:

"**Exploiting the Table of Energy and Power Leverages**", Issam Raïs, Anne Benoit, Laurent Lefèvre and Anne-Cécile Orgerie, *ICA3PP: International Conference on Algorithms and Architectures for Parallel Processing*, Guangzhou, China, pages 175-185, November 2018.

"**Building the Table of Energy and Power Leverages for Energy Efficient Large Scale Systems**", Issam Raïs, Mathilde Boutigny, Laurent Lefèvre, Anne-Cécile Orgerie and Anne Benoit, *HPCS: International Conference on High Performance Computing & Simulation*, Orléans, France, pages 284-291, July 2018.

"**Experimental analysis of vectorized instructions impact on energy and power consumption under thermal design power constraints**", Amina Guermouche and Anne-Cécile Orgerie, *research report*, pages 1-11, June 2019.

This concludes our section on solutions based on reducing the idle energy consumption of Cloud infrastructures. Fighting against non-power proportionality of Cloud equipment stays an unavoidable solution for not fully utilized data centers. We showed that switching off policies have a bright future, even with heterogeneous envisioned computing architectures. In parallel to this work, I conducted research on rethinking the Cloud architecture to make it more energy-efficient on the whole.

III.C Redesigning Cloud architectures

With the emergence of personal mobile devices, a growing amount of data is being generated and consumed everyday. These data occupy data centers that can be located far away from where data are needed. This situation is especially intense in the case of geographically constrained information. In many cases, the geographical distance between clients is very small compared to the distance between the clients and the data processing and storage servers of centralized Clouds. Whilst, by design, this situation is supported by Cloud computing, the existing implementations, employing large centralized data centers, become a bottleneck when it comes to latency, network flooding and the provision of resources. Networks constitute the key elements interconnecting the data centers and the users. However, the network devices present even less non-power proportional profiles than servers (Section II.B.2).

During the IC0804 COST Action[2], we had a focus group on *green wired networks*. We explored techniques to improve the energy efficiency of wired communication networks, from access networks to core networks. This work was published in a book chapter in:

[2] IC0804 COST (European Cooperation in Science and Technology) Action on Energy efficiency in large scale distributed systems (2009-2013) `http://www.cost804.org`

"**Green Wired Networks**", Alfonso Gazo Cervero, Michele Chincoli, Lars Dittmann, Andreas Fischer, Alberto Garcia, Jaime Galan-Jimenez, Laurent Lefèvre, Hermann de Meer, Thierry Monteil, Paolo Monti, Anne-Cécile Orgerie, Louis-Francois Pau, Chris Phillips, Sergio Ricciardi, Rémi Sharrock, Patricia Stolf, Tuan Trinh, and Luca Valcarenghi, *chapter in Large-Scale Distributed Systems and Energy Efficiency: A Holistic View*, pages 41-80, Wiley Series on Parallel and Distributed Computing, John Wiley & Sons (ISBN 978-1-118-86463-0), April 2015.

Several issues occur when dealing with wired communication networks: the solution needs to be global and interoperable, it should preserve the overall connectivity of the network (i.e. any pair of nodes should be able to communicate at all time), and it should not impact noticeably the quality of service (i.e. latency, bandwidth and jitter). Despite these troubles, switching off unused network devices seems an interesting idea from an energy-efficient point of view. This work started with the thesis of Ismael Cuadrado Cordero (the first PhD student that I co-advised, with Christine Morin, between October 2013 and February 2017), at a time when *edge clouds* and *fog computing* were not spread in the Cloud research community.

III.C.1 Network-aware Cloud infrastructures

The work presented hereafter has been published in:

"**GRaNADA: A Network-Aware and Energy-Efficient PaaS Cloud Architecture**", Ismael Cuadrado Cordero, Anne-Cécile Orgerie and Christine Morin, *GreenCom: IEEE International Conference on Green Computing and Communications*, Sydney, Australia, pages 412-419, December 2015.

As shown in Section II.C.1, network devices can weigh heavily in the overall energy consumption of a distributed Cloud infrastructure with data centers geographically spread. This vision of split resources opposes to the original centralized cloud implementation, where servers are located in the same large data centers. In a centralized approach, according to our measurements, an average French user would need to go through 12 different hops (level-3 network devices), before being connected to the internal cloud network. If the same user is connecting from the USA, it would take only 2 hops to access the same service. Once inside the cloud's network, data are sent from and to different data centers locations according to availability and contextual factors. This is the case of services like Google Drive [Goo], where two French users working over the same document will have, on average 20 hops between them (10 hops each to the Irish Google's data center for instance). In many cases, information is shared among users located in similar geographical regions [CMG09]. In this context, the use of a centralized system might cause unnecessary delays and packet forwarding outside the network.

On the other hand, while fully distributed solutions provide great robustness and low latency, they fail to provide simultaneous modification accesses to files [JAV+14, LEGE14]. Moreover, due to replication of content, the use of decentralized cloud systems require a greater bandwidth utilization, as well as additional energy expenses. In the example of online document edition, the two users would be modifying their own copies of the same file, facing merging conflicts in case of concurrent utilization. Consequently, in order to keep synchronization of data, a vast flow of information should be continuously exchanged between clients. If the number of participants accessing the document is too large, the required bandwidth might imply the utilization of several paths. Having all these paths on might make the peer-to-peer approach less energy-efficient than the centralized one.

We bet that the future of cloud computing relies on a better geographical distribution of resources for improving performance and energy-efficiency. Towards this end, we propose the concept of *microcloud*, a fully autonomous energy-efficient sub-network of clients of the same service, designed to keep the greenest path between them. A microcloud can be seen as an autonomous set of clients, among which a Light Virtual Machine (LVM) is deployed on one of them. The LVM is a partial version of a VM containing only the data needed by the clients in the microcloud. It is accessed by the clients belonging to the same microcloud.

This system targets services where the geographical distribution of clients working on the same data is limited - for example, a shared on-line document - or services where, even if the geographical distribution of clients is high, the upload data communication to the cloud is small - for instance a light social network like Twitter. Microclouds rely on a cloud-aware routing protocol, named DEEPAC, that distributes the communication between nodes in the network. The underlying idea consists in keeping the Cloud traffic as low as possible and switching off unused network equipment. Network devices being even less power proportional than servers (as detailed in Section II.B.2), switching off remains the easiest option to save large amounts of energy (as shown for servers in Section III.B.1). However, switching off network devices require to carefully reroute the network traffic through other paths with switched-on devices. DEEPAC ensures that between any two user devices of the microcloud, a switched-on route exists at all time (without fault-tolerance though).

For each microcloud, a manager controls the access by new clients and the security of the application, communicates with the Cloud data center for backup purpose, and splits the microcloud if the number of devices reaches a given upper limit. Each microcloud also comprises a provider that runs the LVM, which contains the application and all the data accessed by the clients. In Figure III.4, a scheme of microclouds interconnection is shown. The vertical communication of microclouds between managers is used as a tunnel to communicate with data centers.

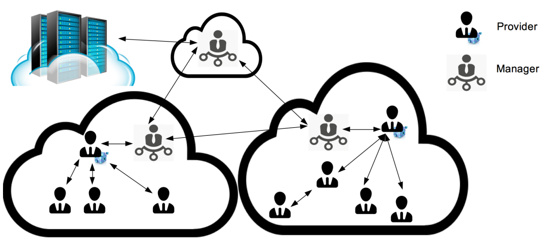


Figure III.4 – Scheme of microclouds interconnection

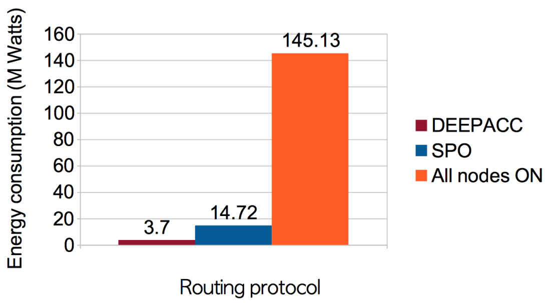


Figure III.5 – Energy consumption of the entire Cloud network with one microcloud under different management protocols

Using the ns3 ECOFEN module presented in Section II.D.1, we evaluated our proposition against centralized cloud architecture from a network energy consumption point of view. Figure III.5 shows the energy consumption of three protocols, assuming all the microcloud clients are 1 hop distant from each other. The *All ON* approach keeps all network devices and links in an active state (current situation). The *SPO* (Shortest Path Only) approach is a fully informed version of OSPF (Open Shortest Path First) [IET98]. It starts from a fully shut down network and, for every node, before starting any communication, calculates the shortest path between the sender and the receiver. Our approach *DEEPACC* starts from a fully shut down network and, for every client node, before starting any communication, calculates the shortest path between the sender and the closest node in the microcloud to minimize the number of switched-on devices.

In Figure III.5, in the case of All ON, all the devices in the network are working and responsive. In the case of SPO, only the devices in the working path are powered on, thus consuming almost 90% less energy than All ON. Finally, DEEPACC outperforms SPO from an energy-efficient point of view, consuming 75% less. However, giving access to a new user takes more time as it requires to compute her route.

This first step strengthens the idea of exploring more decentralized Cloud architectures for saving energy. Our microcloud architecture would now belong to the fog computing category, while it was not yet defined at the time of this work. Since then, numerous amounts of work have been proposed on this subject, presenting contributions close to ours [MBM+18, AS17, YLH+18].

III.C.2 Towards energy-efficient mobile edge clouds

Going a step further in the decentralization of Cloud infrastructures, we adapt the concept of microclouds to a smart city context to provide a platform for mobile Cloud computing. To do so, local microclouds are created by merging static public devices, such as the smart city infrastructure and networking equipment belonging to the Internet Service Provider, and private static and mobile devices (i.e. computers and the users' mobile devices). We consider these devices to be located across a given bounded geographical area, typically a neighborhood in a city. Microclouds provide the smart city infrastructure with lightweight mechanisms to handle the dynamism of a mobile edge Cloud. Users may arrive at or leave the considered geographical area, as well as move inside the boundaries of the neighborhood. Also, it eliminates the need for dedicated infrastructures (i.e. datacenters) and provides a dynamic and tailored environment where multiple services coexist. The work presented hereafter has been done in collaboration with Queen Mary University of London (UK), and published in:

"**Microcities: a Platform based on Microclouds for Neighborhood Services**", Ismael Cuadrado Cordero, Felix Cuadrado, Chris Phillips, Anne-Cécile Orgerie and Christine Morin *ICA3PP: International Conference on Algorithms and Architectures for Parallel Processing*, Granada, Spain, pages 192-202, December 2016.

"**Microcities: a Platform based on Microclouds for Neighborhood Services**", Ismael Cuadrado-Cordero, Felix Cuadrado, Chris Phillips, Anne-Cécile Orgerie, Christine Morin, *research report*, RR-8844, 17 pages, 2016.

The basic idea is to use microclouds as a support infrastructure for mobile devices to offload computation. Offloading computation in a nearby cloud infrastructure allows mobile devices to utilize application with low-latency requirements. Applications targeting the specific population of a neighborhood are a good example of geographically localized services. From a platform perspective (i.e. the deployment of neighborhood applications), many services are only of interest to the population of a community. For instance, information about street works, water or electricity cuts or local store information, such as goods in stock or opening hours, affect only neighbors of the area, who benefit from these utilities.

In order to show the viability of using microclouds in a real-life environment, we built a prototype using 10 nodes in a Local Area Network (LAN). These experiments have been used to obtain real-life data about latency and packet loss probability. The infrastructure is deployed as follows. We use 10 nodes, 6 laptops (4 MacBook Pro 2.7 GHz Intel Core i7 8 GB 1600 MHz DDR3, 2 HP EliteBook 2.10GHz Intel Core i7-4600U CPU 16GB 1600 MHz DDR3), 1 multipurpose small computer (Raspberry Pi 2 model B 900MHz quad-core ARM Cortex-A7 CPU 1GB RAM), 2 smartphones (OnePlus One Qualcomm Snapdrago 801 processor with 2.5GHz Quad-core CPUs running CyanogenMod 11S based on Android 4.4 and iPhone6 Dual-core 1.4 GHz Typhoon ARM v8-based running iOS8) and 1 network switch to which all are connected (DELL PowerConnect 6224). Connections use WiFi for the case of smartphones and Ethernet in the case of laptops and multipurpose small computer. Among the computers, one has been chosen as a service provider, and the rest as clients. We compare the performance of this configuration against a scenario where the service provider is located on a VM hosted in Amazon Cloud service (world-wide area).

The communication process simulates the interaction between clients and sever in an on-line shared document application. Given the lack of traces in literature for concurrent access to multiple users' Cloud applications, we have obtained several real 45-minutes traces from actual Google Drive sessions, using the network packet analyzer tool Wireshark [Wir]. These traces are obtained from the concurrent use of documents (addition/deletion of text). We first compare the communication delay perceived by the users, and as expected, our approach provides a latency several times smaller than using a datacenter-centralized solution in very localized environments. Indeed, the microcloud obtains an average delay of about 15 ms between clients, while the centralized experiment shows an average RTT of about 117 ms. Results are explained by the distance between users in the case study (almost negligible compared to Amazon's world-wide area). Second, we randomly changed the location of the service provider in the microcloud, thus forcing our prototype to redeploy the LVM. Once it is deployed, the former service provider (the node previously running the server software) sends a migration message to all the nodes with the new IP. Finally, all nodes start the communication process with the new service provider. This process was done 10 times. We did not register packet loss during these experiments. This is explained because the protocol used in the connection is TCP, which ensures the arrival of the packets at destination. Also, all clients are aware of the change once it is available.

We then extrapolated the obtained data to a larger network. To do so, we simulated a synthetic physical neighborhood topology using ns3 [ns3]. The simulator reproduces the whole communication process up to a packet level using the traces captured during our prototype evaluation. Given that ns3 is a packet level simulator, simulating mobile nodes takes a long time (simulating more than 100 mobile nodes moving over the static infrastructure takes almost 24h to represent 1 hour worth of users interaction). The simulated network contains a variable number of mobile nodes (between 2 and 100), with a random mobility over a physical network of 45 static nodes, which represents the smart city infrastructure.

Figure III.6 shows the probability of migrating a LVM using either one large or several smaller microclouds' configurations. As a comparison, Figure III.7 shows the number of devices occupied in the network: a node is occupied if it is part of, at least one microcloud. This gives a rough evaluation of the energy consumption of the overall infrastructure: the more nodes it uses, the higher its energy consumption. It can be observed that when the number of occupied nodes increases, the number of migration decreases. This is explained by how DEEPACC behaves in case of reconnection. As explained in Section III.C.1, when a mobile node connects to a microcloud DEEPACC finds the shortest path between the mobile node and any node in the microcloud. Thus, in the case of a reconnection, a mobile node disconnects from a static node and connects to a new one. In this situation, if this new node was already part of the former microcloud, it is possible that other mobile nodes are already connected to it. Then, latency to the mobile node is the same as the latency to its neighbors, which was considered acceptable. However, if the static node to which the mobile one connects does not participate of the microcloud before, then the added latency may be much higher, since more devices may add more latency.

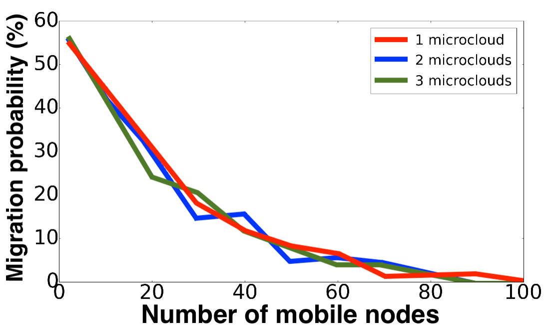


Figure III.6 – Probability of migration of LVM

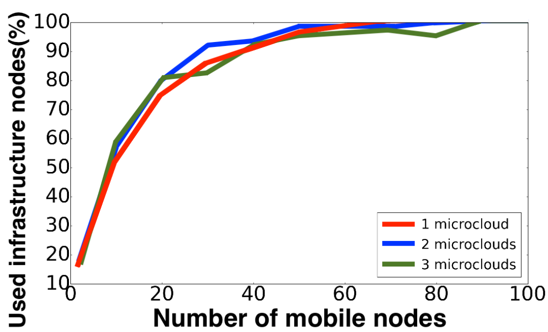


Figure III.7 – Utilization of the network

This usecase illustrates that microclouds are able to exploit network resources to reduce unnecessary data transportation over long distance networks, running computation on the nodes participating in the communication, like personal devices, network equipment and/or specific purpose hardware such as smart cities networks. We show the benefits of using our architecture over the currently dominant datacenter-centralized approach in terms of quality of service and node utilization. While our system requires extra management computation from participating nodes, we showed in the original paper that this overhead is significantly reduced due to its adaptability and the use of a threshold-based heuristic.

Distributed Cloud infrastructures, such as Edge and Fog, have a clear advantage in terms of quality of service for the users (i.e. low latency). Yet, on the provider side, it may be less profitable as it increases the maintenance costs for instance. We performed a cost-benefit analysis of centralized and distributed architectures from a financial point of view to compare them. This work, not detailed in this manuscript, has been done during the post-doc of Anthony Simonet, and published in:

"**Deploying Distributed Cloud Infrastructures: Who and at What Cost?**", Adrien Lebre, Anthony Simonet and Anne-Cécile Orgerie, *Workshop on Cloud Computing Interclouds, Multiclouds, Federations, and Interoperability (Intercloud)*, in conjunction with IC2E, Berlin, Germany, pages 178-183, April 2016.

We also studied the utilization of non-lossy compression to reduce the size of data transfers and their subsequent impact on the networks. This work, not detailed in this manuscript, has been done in collaboration with Rutgers University (USA) within the context of the SUSTAM project[3], and published in:

"**Leveraging energy-efficient non-lossy compression for data-intensive applications**", Issam Raïs, Daniel Balouek-Thomert, Anne-Cécile Orgerie, Laurent Lefèvre and Manish Parashar, *HPCS: International Conference on High Performance Computing & Simulation*, Dublin, Ireland, July 2019.

From the infrastructure side, improving the energy-efficiency of distributed Clouds through a redesign of their architecture can lead to consequent energy savings in the case of application with highly localized traffic. On the software stack side, solutions also exist to chase energy waste.

III.C.3 Improving the energy-awareness of Cloud management stacks

The separation of the cloud stack in two distinct IaaS and PaaS layers, while having great advantages for portability and separation of concerns, can be detrimental in terms of energy awareness. The work presented hereafter has been published in:

"**Towards Energy-Aware IaaS-PaaS Co-design**", Alexandra Carpen-Amarie, Djawida Dib, Anne-Cécile Orgerie and Guillaume Pierre, *Smartgreens: International Conference on Smart Grids and Green IT Systems*, Barcelona, Spain, pages 203-208, April 2014.

If each Cloud layer is allowed to take energy-related decisions independently, these uncoordinated actions can lead to significant resource waste and performance degradation, possibly negating the benefits of energy awareness altogether. For instance, the IaaS layer can decide to migrate a virtual machine (VM) in order to perform a better server consolidation for energy-efficiency purposes. Yet, this same VM may end a few seconds later because it gets released by

[3]Inria associated team SUSTAM on Sustainable Ultra Scale compuTing, dAta and energy Management (2017 - 2019) `https://graal.ens-lyon.fr/sustam/`

the PaaS layer. The decision to shutdown this VM may have been taken several minutes in advance by the PaaS layer. If this information is not communicated to the IaaS layer, we take the risk that IaaS will invest previous resources (for example by migrating the VM) without seeing any benefit from this action (because the VM gets shut down just after).

Conversely, the PaaS layer may help the IaaS layer in performing its VM management actions. For example, it is often easy at the PaaS level to temporarily redirect one VM's workload to another (by redefining load balancing parameters for example). Offloading a VM for just a few tens of seconds may greatly facilitate IaaS-level management tasks such as VM migration.

In order to avoid counterproductive independent optimizations, IaaS and the PaaS could share their energy-related information and coordinate their reconfiguration actions. This coordination aims at allowing system-level optimizations and trade-offs. To facilitate the interaction between cloud layers while preserving the separation and the interoperability across the cloud stack, we argue there is a need for an abstraction layer proposing coordination APIs. Such a mechanism is required to deploy distributed architectures such as the microcloud approach presented in III.C.1.

This concludes our section on redesigning Cloud architectures for energy-efficiency purpose. The development of an efficient software stack for managing distributed Clouds is a hot topic in the community This was the main focus of the Discovery project[4], and we discussed this point in a book chapter in:

"**Beyond the Clouds: How Should Next Generation Utility Computing Infrastructures Be Designed?**", Marin Bertier, Frédéric Desprez, Gilles Fedak, Adrien Lebre, Anne-Cécile Orgerie, Jonathan Pastor, Flavien Quesnel, Jonathan Rouzaud-Cornabas, and Cédric Tedeschi, *chapter in Cloud Computing - Challenges, Limitations and R&D Solutions*, pages 325-345, Springer (ISBN 978-3-319-10529-1), November 2014.

Providing APIs to enable the communication of energy-related information between the Cloud layers could help in reducing the overall energy consumption. While such an API seems interesting between IaaS and PaaS layers, one could go even upper in the layers, and ask directly to the users for their help.

III.D Involving Cloud users in energy savings

At first, it seems not profitable to ask users and customers to use less the Clouds' resources that they rent. To reduce the electrical consumption of cloud infrastructures, consolidation mechanisms pack the virtual machines (VMs) on the least number of servers, without impacting application performance, in order to turn off the unused servers in case of moderate load. Idle servers indeed consume extensive amounts of energy as shown in Section II.B.1. However, such consolidation techniques are only efficient if virtual resources are not kept idle by the users for no work. Indeed, if the cloud provider does not over-commit the physical resources, the user that employs only partly the VMs resources is wasting the rest. To be energy-efficient, users need to properly size their VMs.

For a given application, several VM sizes are possible, each offering a different trade-off between the overall energy consumption and the performance (i.e. runtime). This trade-off is complex to determine: small-sized VMs may be easier to pack into server machines, while larger VMs may end their work faster. While it is logical that well-dimensioned VMs are more energy efficient, defining their size is not an easy task for the users.

[4] Inria project lab aiming at designing a DIStributed and COoperative framework to manage Virtual EnviRonments autonomicallY (2015 - 2019): `http://beyondtheclouds.github.io`

Beyond the VM size, users can also offer other means of flexibility depending on their use case, such as delaying their VM allocation, or allowing for pause/resume cycles at given time with adequate counterparts (e.g. financial). Furthermore, increasing the interactions between Cloud systems and Cloud users around energy management issues could increase the energy-awareness on both sides. This work was in particular developed during the PhD thesis of David Guyon (September 2015 - December 2018) that I co-advised with Christine Morin.

III.D.1 Proposing users VM sizes options

The work presented hereafter has been done in collaboration with Lawrence Berkeley National Laboratory (USA) in the context of the Dalhis project[5], and published in:

"**Energy-efficient User-oriented Cloud Elasticity for Data-driven Applications**", David Guyon, Anne-Cécile Orgerie and Christine Morin, *GreenCom: IEEE International Conference on Green Computing and Communications*, Sydney, Australia, pages 376-383, December 2015.

"**How Much Energy can Green HPC Cloud Users Save?**", David Guyon, Anne-Cécile Orgerie, Christine Morin and Deb Agarwal, *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, Saint Petersburg, Russia, pages 416-420, March 2017.

"**Involving Users in Energy Conservation: A Case Study in Scientific Clouds**", David Guyon, Anne-Cécile Orgerie, Christine Morin and Deb Agarwal, *International Journal of Grid and Utility Computing*, Inderscience, volume 10, no. 3, pages 272-282, May 2019.

In this work, we propose a cloud system involving users in the energy optimization system. This work is evaluated through the use of real applications that are scientific workflows. A user who agrees to reduce her impact on the environment can choose a more energy-efficient execution mode, implying a loss in performance, by executing her application on less resources on the infrastructure. The unused resources are free for other applications and thus, this approach favors a better consolidation of the whole system. The better the consolidation, the lower the electrical consumption. The proposed system offers three execution modes: *Big*, *Medium* and *Little*. An algorithm selects the size of the VMs for executing each task of the workflows depending on the selected execution mode. The Medium mode executes using the user-specified VM resources for each workflow stage. The Little and Big modes respectively decreases or increases the VMs by one size for the whole workflow.

We employ three scientific applications from different scientific areas that exhibit different behaviors in terms of resource consumption: disk-intensive, CPU-intensive and memory-intensive:

- Montage is an engine to build astronomical image mosaics for astronomers [Cal]. This workflow is mainly IO-intensive and CPU-intensive during the calculation.
- Blast is a program that compares nucleotide or protein sequences to sequence databases and calculates the statistical significance of matches [NCB]. The execution of the workflow has a cyclic use of the memory and constantly uses the CPUs, making it a memory-intensive and a CPU-intensive application.
- Palmtree is a library for the parallelization of Monte Carlo methods where the challenge is the proper management of the random numbers [Len16]. The workflow structure is composed of 2 parallel tasks and its execution is CPU-intensive only.

The three applications are executed on servers of the taurus cluster of Grid'5000 [BCAC+13]. A summary of the execution time versus the energy consumption of each workflow in each execution

[5] Inria associated team on Data Analysis on Large-scale Heterogeneous InfrastructureS (2013 - 2018) `https://project.inria.fr/dalhis/`

mode is given in Figure III.8. The number of servers required to run the workflows increases when the Big mode is selected which explains the energy consumption increasing. On the other hand, the execution time increases by a factor of 3 and more when the Little mode is selected. Users have to find their own trade-off between performance and energy consumption.

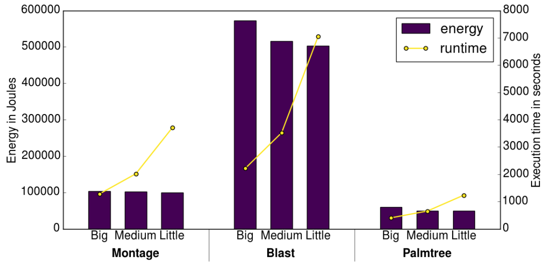


Figure III.8 – Energy consumption and execution time of each workflow in each execution mode.

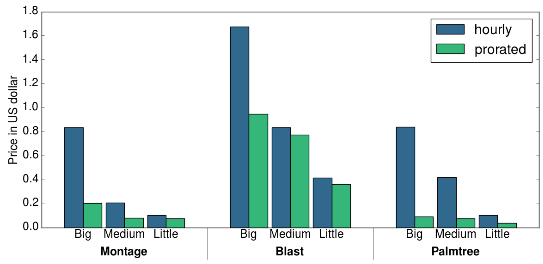


Figure III.9 – EC2 hourly pricing and the prorated pricing of each workflow in each execution mode.

Figure III.9 gives an idea about how much it would cost to run these workflows on the Amazon EC2 platform. On this platform users pay the access to their instances by hour even if the instances are not used a complete hour. The figure presents the EC2 hourly pricing but also the price if a prorated pricing were available (based on utilization time). It shows that the Big execution mode costs more than the Medium mode that also costs more than the Little mode.

Encouraged by these first results, we then evaluate the impact of the proportion of users selecting the Big, Medium or Little mode on a data center's energy consumption. Our evaluations have been done using the same three scientific workflows, the energy consumption measurements for the execution of these workflows on the taurus nodes of Grid'5000, and real traces of jobs submitted to a production HPC center located in the Czech republic [FTK14].

Table III.2 – Energy consumption of a whole cluster used during 24h for various profiles of execution modes

Big	Medium	Little	Energy (kWh)	Std dev energy	Hosts used	Std dev hosts	Energy saved
100	0	0	632.489	16.277	282	7.909	0.00 %
100	0	0	292.941	3.690	292	16.806	53.68 %
0	100	0	234.122	4.882	168	6.363	62.98 %
0	0	100	231.921	3.840	143	3.187	63.33 %
80	0	20	273.205	6.021	236	16.117	56.80 %
60	0	40	269.969	3.497	208	11.071	57.32 %
40	0	60	258.138	3.980	190	14.935	59.19 %
20	0	80	246.996	3.701	170	6.610	60.95 %
20	20	60	246.590	5.482	167	9.843	61.01 %
20	60	20	242.464	4.013	171	9.243	61.67 %

Table III.2 presents our simulation results. We simulate a full day and a cluster with 330 servers (minimum number of servers required to be able to respond to the demand in the highest demand peak). Each row presents the results for a profile distribution following the percentages given in the 3 first columns. All results are the average of 10 simulations and contain the energy consumption in kWh of the whole cluster, the maximum number of hosts required to execute the workload and the standard deviations.

The gray row of the table corresponds to a simulation on a usual cloud infrastructure without any energy optimization. The unused servers are not powered down and all users select the Big execution mode, since it reflects a common behavior when users want results as soon as possible. The last column in both tables is the percent of energy saved compared with the scenario of the first row. A scenario with a 50% energy saving means its execution consumes half compared to the execution with the scenario of the first row.

In a realistic situation, users will not be 100% using the same execution mode but will rather exhibit various behavior. For table dimension reasons, Table III.2 does not contain all possible distribution configurations but still reveals a link between the user profiles and the energy consumed. It shows promising energy savings when the amount of users selecting the Big mode is low. It also shows that using the Little mode compared to the Medium mode does not always provide the best performance/energy saving trade-off.

In this work, we explore a way for energy-aware cloud users to reduce their energy consumption on cloud infrastructures by reducing the size of their virtual machines. We study the influence of energy-aware users on the system energy consumption and compare it with the consumption of more aggressive users in terms of resource utilization. But, VM size is not the only way for Cloud users to help in saving energy. In all cases, information should be provided to the user in order to help her deciding which trade-off she wants between performance and energy metrics.

III.D.2 Playing on VM allocation with the users' agreement

As explained in Section III.C.3, energy savings are possible when enhancing interactions between IaaS and PaaS layers. IaaS knows about the availability of hardware resources and can deliver energy-related information that could help the PaaS layer to make energy-aware decisions. In return the PaaS layer could inform IaaS providers on users' applications flexibility in order to help the consolidation process. The work presented hereafter has been published in:

"**Energy-Efficient IaaS-PaaS Co-design for Flexible Cloud Deployment of Scientific Applications**", David Guyon, Anne-Cécile Orgerie and Christine Morin, *SBAC-PAD: International Symposium on Computer Architecture and High Performance Computing*, Lyon, France, pages 69-76, September 2018.

Typically, when PaaS users ask for virtual resources, requested resources are made available as soon as possible. However, some users could accept their request to be handled differently if it saves energy by either delaying the deployment or changing the VM size (and consequently the duration and price as shown in Section III.D.1). While this approach is not compatible with applications that continuously execute (e.g. web jobs), time-bound scientific applications could exhibit flexibility on starting time and resource size as long as results arrive before a deadline and if the total cost does not increase. This scenario is realistic as scientists running HPC applications are more and more looking at clouds as a cost effective alternative to HPC [NCR+18].

In this work, our objective is to reduce IaaS datacenter energy consumption with a cooperation allowing PaaS to express the flexibility of its applications and IaaS to inform on when and how many resources are predicted to be unused. This way, energy savings are achievable by shifting and resizing some applications on these otherwise unused resources, with the user's agreement.

Figure III.10 presents the general idea on a toy example. It shows three possible allocation options for an initial request of a VM with 8vCPUs on a platform with 2 servers and 3 VMs already allocated on the first one. Starting the application at submission time (C1) requires to turn on server 2. Delaying (C2) or changing the size (C3) can avoid the need of server 2, thus saving energy.

The complete system architecture is presented in Figure III.11. The steps are as follows:

1. a user sends a request to the PaaS provider to execute her application;
2. the PaaS provider sends several requests adjusted to the user flexibility to multiple IaaS providers;
3. each IaaS provider proposes an execution contract (spatial and temporal placement of a VM on servers that stays within the deadline requested by the user) for each request it received;

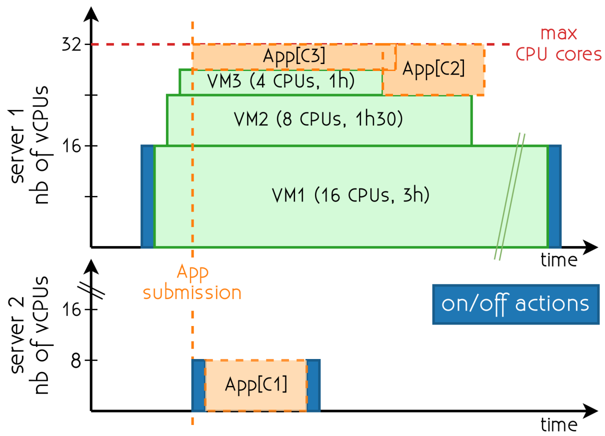


Figure III.10 – Possible scheduling of an 8 vCPUs application in an infrastructure with 2 servers and 3 VMs.

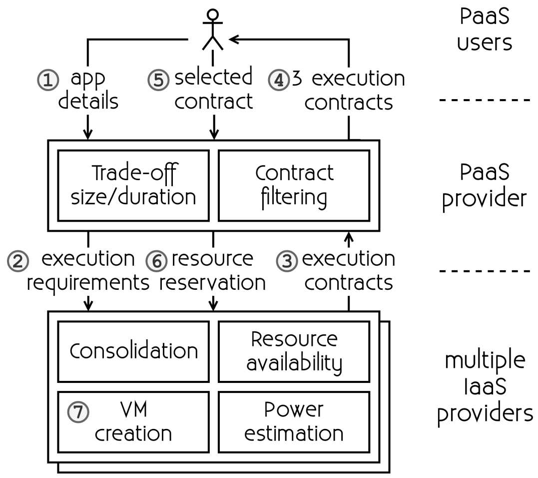


Figure III.11 – Detailed system architecture with the components of both cloud layers and the interactions between them and the end-user.

4. the PaaS provider filters the IaaS propositions based on their energy cost and delay in order to propose three contracts to the user;
5. the user selects the contract she wants to execute her application with;
6. the PaaS provider informs the corresponding IaaS provider that its contract has been selected;
7. this IaaS provider plans the execution of the application in a VM as defined in the contract.

Our evaluation by simulation, presented in the original paper, is based on real data and expresses a large scale cloud scenario. Results show that according to the proportion of energy-aware users, this system is able to reduce the amount of servers by using resources that would have been wasted otherwise. Therefore, our solution allows datacenters to consume less energy than with usual resource managers where all applications start their execution at submission time with their initial resource size. In addition, up to 5.49% of energy is saved compared to a scenario already consolidating the workload, powering down unused servers and where all users prefer their application execution to start at submission time. This result demonstrates that adding flexibility on the time allocation (i.e. delaying the execution) increases the energy savings.

Our proposition of PaaS-IaaS co-design offering to PaaS users energy-efficient execution trade-offs allows them to decide the flexibility level they agree to give to their execution. Financial cost can be a powerful incentive to opt for less energy consuming options. Yet, if this parameter does not balance for the least consuming option, other means should be deployed to motivate users to change their trade-off.

III.D.3 Incentivizing Cloud users to help for energy efficiency

Users can play a key role in saving energy in Cloud infrastructures. Yet, if they do not get a benefice out of it, approaches relying on users' energy-awareness will probably stay at the vain wish stage. The work presented hereafter has been published in:

> "**Incentives for Mobile Cloud Environments through P2P Auctions**", Ismael Cuadrado Cordero, Anne-Cécile Orgerie and Christine Morin, *CloudNet: IEEE International Conference on Cloud Networking*, Pisa, Italy, pages 248-253, October 2016.

This issue is even more severe in the context of mobile Cloud computing considered in Section III.C.2 (when deploying microclouds in the context of smart cities). Mobile Clouds are collaborative decentralized infrastructures which allows mobile devices to unload computation to a local Cloud formed by mobile and static devices. This kind of Cloud provides a better service to latency sensitive applications, due to its physical proximity to the VM host. However, in these systems, the problem of free riding users becomes more acute, for the heterogeneity of devices (from smartphones to private servers) makes the gap of contributed resources much larger. In this work, we analyze the use of incentives for Mobile Clouds, and propose a new auction system adapted to the high dynamism and heterogeneity of these systems.

Mobile clouds may be unfair when some users contribute more resources than others. The unfairness associated with the free riding problem represents an obstacle to the adoption of a collaborative technology. To incentivize users to share their devices and thus to save energy for the entire system, we propose to extend the concept of lease. A lease is a contractual arrangement between an entity, which rent part of its computational power, and a group of users, which offers a payment in return. In a mobile cloud lease, the entity offering the computational power is formed by a group of users, called sellers. The sellers rent part of their resources to host the Virtual Machines (VMs) used to provide the service. On the other hand, the rest of users, called buyers, pay the sellers for hosting the VM. As a consequence to this new concept of lease, a pricing system is required.

Existing solutions are mainly based on two leasing models: fixed and negotiated pricing. In a fixed price system, a seller offers its resources at a specific cost, and the buyers match it. In a negotiated price system, the price of the resource is established by direct competition (auction) between buyers and sellers. In this work, we propose a multi-sided auction system, where the user becomes both buyer and seller, auctioning on other users as needed. Furthermore, we propose an open auction system where the application provider supervises the process, and has the possibility of bidding along with one or more users if the expected result of the auction is unfair to other users.

We compare our solution to other existing auctions systems through simulations with ns3 under the scenario described in Section III.C.2 with 45 static nodes and up to 100 nodes moving randomly. On this scenario, we test several bidding strategies and we evaluate the clients' satisfaction in each case. We defined the satisfaction of a node as the difference between what it required and what it gets, similar to other works such as [WTM14, STM14]. Our simulation results show that the proposed auction-based mechanism performs well in all the tested situations and ensures an acceptable level of satisfaction for all the users. This highlights the suitability of our proposition.

Double-auctions are a commonly accepted incentive system in literature. However, double-auction systems do not always provide a fair incentive system in a highly-dynamic and multi-user scenario such as mobile clouds. Our approach is able to provide a better solution in these scenarios than existing ones. We automate a process in which the owner of the service (which, in the end, is the most interested party in the success of the service) has the ability of injecting external credit in the system to avoid abuse of power from wealthy users. We show that this injection of extra credit benefits the competitiveness of the system, as in our simulations, more credit circulating in the system implies a fairer distribution of credit between users. Since this work, numerous ones have been proposed in this direction: involving auction-based mechanisms for managing users' device participation to mobile edge clouds [YLH+18, CJLF16, JSW+16].

Another indirect way to incentivize users to opt for a less consuming option that could delay their executions, consists in meeting deadlines, negotiated with the users, as in the solution proposed in Section III.D.2 to delay the execution of VMs. The Cloud manager can also boot switched-off resources in advance, to shorten the users waiting time for available resources. For both solutions, Cloud providers have to predict the Cloud workload, otherwise they risk losing energy instead of saving it. Achieving good predictions is then crucial.

This concludes our section on involving users in energy-efficient mechanisms for managing cloud resources. Users' cooperation through incentive mechanisms can achieve consequent energy savings. More broadly, combined with adequate metrics and ecolabels, it could also raise the users' awareness of their own energy impact on Internet infrastructures.

III.E Perspectives

This chapter summarizes my contributions to improving the energy-efficiency of distributed infrastructures. From switching off resources, to redesigning Cloud infrastructures, and involving users, I followed diversified research directions, some being already largely explored and others, not at all. In both cases, I followed the risky tracks presented in this chapter principally along with three PhD students: Ismael Cuadrado Cordero (currently research engineer at Atos in Sevilla, Spain), Issam Raïs (currently post-doc at the Artic University of Norway) and David Guyon (currently post-doc at Inria in Nantes).

Right-sizing infrastructures. Since Internet and cloud computing infrastructures are still far from power-proportionality, dynamic adaptation methods attempt to increase the energy efficiency of these existing systems. Such power management techniques comprises Dynamic Voltage and Frequency Scaling (DVFS) and sleeping states for a given device. At the infrastructure level though, heterogeneous architectures provide an additional lever to better adjust the power consumption and to allocate adequate resources to the users' applications. Dynamic adaptation aims ideally at allocating the exact required amount of resources and power to each application at any time, avoiding resources and power wastage. I will continue to seek dynamically right-sized infrastructures, in terms of both hardware resources and software management layers, in the upcoming context of more and more distributed systems, such as edge and fog computing. Indeed, the geographic dispersion of locally-limited resources adds to the already complex issue of power management. This work will especially be conducted in collaboration with colleagues from Northeastern University, USA, as part of the FogRein project[6].

Capping power. The PUE (Power Usage Effectiveness) metric highlights the still high energy cost of cooling in Internet infrastructures, as detailed in Sections II.C.1 and II.C.4. One way of limiting heat dissipation in data centers consists in avoiding power peaks through power capping techniques. Firstly, these techniques were employed at CPU level to counter the effects of dark silicon [Tay12], and at infrastructure level, to optimize the energy-related financial budget [CHCC13]. Power capping constitutes a promising way of reducing heat dissipation and its relative energy cost, specially on already deployed data centers with air conditioning facilities. Yet, as often with energy and performance metrics, trade-offs are required between the quality of service perceived by users and the power cap constraints ensuring energy savings at the infrastructure level. Finding such trade-offs necessitates analytical models linking power cap values, heat dissipation, energy consumption and application performance. I plan to investigate this technique within the context of the Hac Specis project[7]. Such an exploration in the power- or energy-constrained budget world could provide, in the long term, valuable clues on energy sobriety for the use phase in ICT systems.

Involving users. As shown in Figure I.1, Internet users are part of the equation ruling the ICT's global energy consumption. Informing users about their consumption and incentivizing them to reduce their impact is crucial to decrease the overall expenditure. Although monitoring tools, carbon taxes or comprehensive metrics, among others, head in this direction, they require explanations to be effectively adopted by users. Moreover, as a research scientist, working on energy-efficiency of

[6] Inria associated team FogRein on Steering Efficiency for Distributed Applications (2019 - 2022)

[7] Inria project lab on High-performance Application and Computers: Studying PErformance and Correctness In Simulation (2016 - 2020) `http://hacspecis.gforge.inria.fr`

distributed systems, I consider knowledge transfer as one key aspect of my job. For this purpose, I devote time and energy for popularization purpose around these subjects, mainly through two ways: introductory lectures on green computing in engineering schools (ENS Rennes, Telecom SudParis, IMT-Atlantique Nantes, ENSSAT Lannion, INSA de Rennes, CentraleSupélec Rennes), and popularization science articles, as the following ones:

- "**Sciences du numérique et développement durable : des liens complexes**", Françoise Berthoud, Éric Drezet, Laurent Lefèvre and Anne-Cécile Orgerie *Interstices*, June 2015.
- "**L'épidémie du smartphone : prolifération et dissémination des composants électroniques**", Françoise Berthoud, Éric Drezet, Laurent Lefèvre and Anne-Cécile Orgerie *Interstices*, June 2015.
- "**La déferlante des données**", Françoise Berthoud, Éric Drezet, Laurent Lefèvre and Anne-Cécile Orgerie *Interstices*, July 2015.
- "**Le syndrome de l'obésiciel : des applications énergivores**", Françoise Berthoud, Éric Drezet, Laurent Lefèvre and Anne-Cécile Orgerie *Interstices*, July 2015.

There is nothing like a dream to create the future.
Victor Hugo

IV
Greening distributed infrastructures

IV.A Introduction to renewable energy

The first way to save on electrical bills at a data center level consists in locating it close to where the electricity is generated, hence minimizing transmission losses. For example, Western North Carolina, USA, attracted data centers with its low electricity prices due to abundant capacity of coal and nuclear power following the departure of the region's textile and furniture manufacturing [Gre11]. This region has three super-size data centers from Google, Apple and Facebook with respective power demands of 60 to 100 MW, 100 MW and 40 MW [Gre11].

Other companies opt for greener sources of energy. For example, Quincy (Washington, USA) supplies electricity to data facilities from Yahoo, Microsoft, Dell and Amazon with its low-cost hydro-electrics left behind following the shutting down of the region's aluminum industry [Gre11]. Several renewable energy sources like wind power, solar energy, hydro-power, bio-energy, geothermal power and marine power can be considered to power up super-sized facilities and reduce their carbon footprint.

As their increasing electricity bill also raises environmental issues, Cloud providers resort more and more to renewable energy [Gre17]. In 2016, according to its environmental responsibility report, 100% of the electricity used by Apple-operated data centers came from renewable energy [App17]. In 2011, when Apple started to report on the carbon emissions of their data centers, they were already claiming to reduce them by 56% compared to the case where they would be entirely supplied from the electrical network, whose electricity shows a less environmentally favorable energy mix. Meanwhile, from 2011 to 2016, these carbon emissions should have been multiplied by almost 5 due to the increase in number of Apple-owned data centers [App17]. However, this is not the case as this renewable energy is mostly provided by Apple-owned electricity generation facilities including solar arrays, wind farms, biogas fuel cells, and micro-hydro generation systems [App17].

Although Infrastructure-as-a-Service (IaaS) Cloud providers intensify their part of renewable energy consumption, they often overestimate their use in proportion to the total consumption and consequently, underestimate their dependence on coal [Res16]. Indeed, the intermittent nature of current most commonly-used renewable sources (i.e. sun, wind) causes major challenges. Hence, an ideal Cloud manager should match its energy consumption with the renewable energy production. Yet, these two curves are *a priori* uncorrelated.

This chapter presents my work on enabling Cloud infrastructures to optimize their utilization of renewable energy sources. I started from one data center in Section IV.B. I pursued by proposing solutions leveraging data centers from different locations in Section IV.C. Thirdly, I proposed a cooperation with Smart Grids to handle the geographical distribution of renewable energy sources IV.D. Finally, Section IV.E concludes this chapter and sketches future work.

IV.B Single data center partially powered by on-site solar energy

Besides the ecological impact, the energy consumption is a predominant criteria for Cloud providers since it determines the daily cost of their infrastructure. As a consequence, power management becomes one of the main challenges for data centers and more generally for large-scale distributed systems. As detailed in Section III.C.2, to improve the performance of their cloud and to leverage their available infrastructure, telecommunication operators, deploy small data centers (20 to 50 servers per data center) at the network border, closer to customers. In this recent architecture, by deploying data centers closer to the user, the response time and throughput are greatly improved.

From an energy point of view, these small data centers allow the study of new power supply solutions based on renewable energy, like wind or sun. Using these renewable energy sources can reduce the operating cost but, unfortunately, this kind of energy stays intermittent by nature. To address this problem, two solutions exist for a single data center: investing in heavy expensive battery systems to smooth over the day the renewable energy production, or developing new applications management solutions adapted to the electricity production. We explored both options. This work was done during the PhD thesis of Yunbo Li (October 2013 - June 2017), that I co-advised with Jean-Marc Menaud, within the context of the EPOC: Energy Proportional and Opportunistic Computing systems (2013-2017, funded by the Labex CominLabs).

IV.B.1 Opportunistic scheduling

The work presented hereafter has been published in:

"**Opportunistic Scheduling in Clouds Partially Powered by Green Energy**", Yunbo Li, Anne-Cécile Orgerie and Jean-Marc Menaud, *GreenCom: IEEE International Conference on Green Computing and Communications*, Sydney, Australia, pages 448-455, December 2015.

"**The EPOC project: Energy Proportional and Opportunistic Computing system**", Nicolas Beldiceanu, Barbara Dumas Feris, Philippe Gravey, Sabbir Hasan, Claude Jard, Thomas Ledoux, Yunbo Li, Didier Lime, Gilles Madi-Wamba, Jean-Marc Menaud, Pascal Morel, Michel Morvan, Marie-Laure Moulinard, Anne-Cécile Orgerie, Jean-Louis Pazat, Olivier Roux and Ammar Sharaiha, *SmartGreens: International Conference on Smart Grids and Green IT Systems*, Lisbon, Portugal, pages 1-7, May 2015.

"**Towards energy-proportional Clouds partially powered by renewable energy**", Nicolas Beldiceanu, Barbara Dumas Feris, Philippe Gravey, Sabbir Hasan, Claude Jard, Thomas Ledoux, Yunbo Li, Didier Lime, Gilles Madi-Wamba, Jean-Marc Menaud, Pascal Morel, Michel Morvan, Marie-Laure Moulinard, Anne-Cécile Orgerie, Jean-Louis Pazat, Olivier Roux and Ammar Sharaiha, *Computing*, Springer, volume 99, issue 1, pages 3-22, January 2017.

In this work, we propose to take advantage of renewable energy availability to perform opportunistic tasks. The data center receives a fixed amount of power from the regular electric grid. This power allows it to run the usual tasks. In addition, it is also connected to renewable energy sources (such as windmills or solar cells) and when these sources produce electricity, it is used to run more, less urgent, tasks. In order to achieve this energy-aware resource allocation, we distinguish two kinds of jobs to be scheduled on the data center: the web jobs which represent jobs requiring to run continuously (like web server), and the batch jobs which represent jobs that can be delayed and interrupted, but with a deadline constraint. The second type of jobs are the natural candidates of the opportunistic scheduling algorithm.

We started this work by studying anonymized traces provided by the EasyVirt SME[1]. Theses traces concern a VM hosting provider with 55 servers. The traces stretch from the 25th of March 2014 to the 6th of July 2014. They consist in the logs for real CPU, RAM, network and disk utilization of each server every 90 seconds. They also contain the client's requests for VMs with CPU and RAM sizes, and the submission dates. These traces present a realistic scenario in our context. Figure IV.1 illustrates the average CPU and RAM utilization of all the servers during a week in the data center. Note that the average CPU utilization keeps at a low state and far below the average RAM utilization, thus leading to a consequent waste of resources.

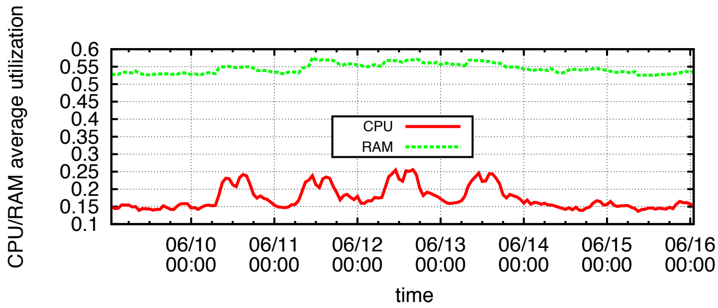


Figure IV.1 – CPU and RAM real utilization over one-week of real trace

To save energy in a single data center, a classical goal is to reduce the number of powered-on servers (as detailed in Section III.B), and the performance of VM placement algorithm directly affects this number. The problem of VM placement is typically modeled as a n-dimensional bin-packing problem which is NP-Hard [CK99]. In 1-dimensional bin packing problem, FFD (First Fit Decreasing) is a classic greedy algorithm which is proved to use: maximum $11/9 \times n + 1$ bins where n presents the number of bins in the optimal solution [Yue91]. In addition, cloud managers resort to use resource over-commitment techniques to increase the resources usage. Our solution will use both techniques: shutdown and over-commitment to reduce the data center's energy consumption.

We proposed PIKA (oPportunistic schedulIng broKer infrAstructure), a framework aiming at reducing the brown energy consumption (ie. from the regular electric grid, assumed to come from non-renewable energy sources), and improving the usage of renewable energy for mono-site data center. It exploits jobs with slack periods, and executes or suspends them depending on the renewable energy availability. By consolidating the virtual machines (VM) on the physical servers, PIKA adjusts the number of powered-on servers in order for the overall energy consumption to match with the renewable energy supply.

As the system is dynamic, PIKA performs the optimization operations periodically. The optimization cycle is defined as a slot, such that the time in our system is divided into a series of continuous slots. At any time, users submit jobs which are VM allocation requests. At the beginning of each slot, the broker executes the three main steps. First, the broker checks each server's state and suspends or migrates some VMs from the overloaded servers (due to over-commitment). Second, the renewable energy predictor predicts the amount of renewable energy for the current slot and informs the broker. The broker then determines the number of servers that can be supported by the renewable energy supply. Finally, according to the available resources from these servers, the broker schedules the jobs that can be executed during the current slot by starting with the (mandatory) web jobs and then the (interruptible) batch jobs. Given an accurate prediction on renewable energy, the broker dynamically switches on and off servers to adjust the energy consumption in order to maximize the renewable energy integration ratio.

To evaluate PIKA, we use the real workload traces from EasyVirt described earlier, real solar energy production traces furnished by the Photovolta project at University of Nantes [Pho], and real server's power profiles measured on Grid'5000 (Taurus cluster). We compare PIKA with a

[1]EasyVirt: `https://www.easyvirt.com/`

baseline allocation policy using a classical FFD algorithm to allocate the VMs. The results of energy consumption for both baseline algorithm and PIKA are shown in Figure IV.2. The top curve presents the baseline result (with FFD algorithm to allocate the VMs) and the bottom corresponds to PIKA.

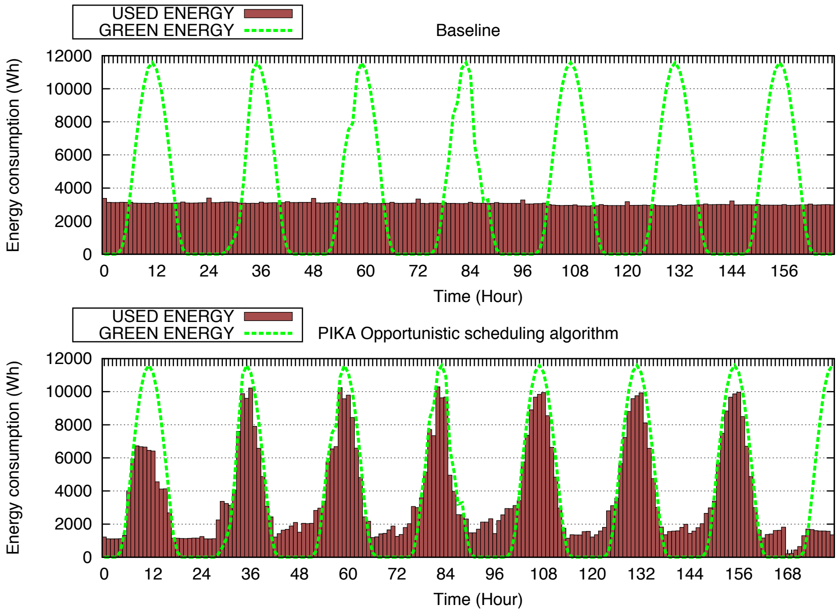


Figure IV.2 – Energy consumption for baseline and PIKA allocation

The energy consumption of baseline is flat. The workload scheduling is not affected by the variable renewable energy supply (the green curve). The energy consumption of PIKA follows the renewable energy variations. PIKA significantly increases the renewable energy integration into data center. While the renewable energy becomes unavailable, the broker switches off some servers and only launches some essential jobs (web job, batch job with approaching deadlines and a few of batch jobs to fill the remaining resources already powered on). Due to its opportunistic behavior, PIKA finishes all the job 11 hours later than the baseline for the entire week execution because some batch jobs are delayed in case of insufficient renewable energy.

Table IV.1 – Energy consumption of the baseline algorithm and PIKA (in kWh)

Algorithm	Total E. C.	Brown E. C.	Renewable E. C.
Baseline	513.633	259.559	254.073
PIKA	676.895	142.957	533.938
	31% ↑	44.9%↓	110.1%↑

Table IV.1 shows the result of brown, renewable and total energy consumption for the baseline and PIKA. Compared to the baseline, PIKA reduces by 44.9% brown energy consumption and increases by 110.1% the renewable energy integration. Yet, the results also indicate that PIKA consumes 31% more energy in total. This is because PIKA performs dynamic VM consolidation to adjust the number of powered-on servers and that leads to a large number of VM migrations compared with baseline (the migration in baseline is only in case of overloaded server). We took an unfavorable scenario where there is no central storage in the data center. Consequently, each server uses its own disks, and VMs have to migrate with their entire disk image whenever they are reallocated or paused (for interruptible batch jobs). Moreover, PIKA requires more time as outlined before to execute the entire workload in order to benefit from opportunistic scheduling. These two factors explain the consequent energy overhead of migrations. Yet, in the case of PIKA, all of this extra energy consumption comes from renewable energy supply.

This work shows the opportunity created by small-sized data centers partially powered by renewable energy in order to save energy for distributed Cloud infrastructures. We proposed a different scheduling algorithm based on constraint programming with colleagues from Nantes. This work, not detailed in this manuscript, has been published in:

"**Green energy aware scheduling problem in virtualized datacenters**", Gilles Madi Wamba, Yunbo Li, Anne-Cécile Orgerie, Nicolas Beldiceanu and Jean-Marc Menaud, *ICPADS: IEEE International Conference on Parallel and Distributed Systems*, Shenzhen, China, pages 648-655, December 2017.

I also contributed, with the same colleagues, to the design of two prediction models (based on constraint programming and neural networks), that were validated against the real workload traces coming from EasyVirt. This work, not detailed in this manuscript, has been published in:

"**Cloud workload prediction and generation models**", Gilles Madi Wamba, Yunbo Li, Anne-Cécile Orgerie, Nicolas Beldiceanu and Jean-Marc Menaud *SBAC-PAD: International Symposium on Computer Architecture and High Performance Computing*, Campinas, Brazil, pages 89-96, October 2017.

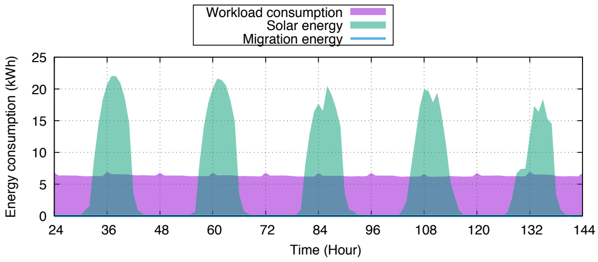
IV.B.2 Batteries

Opportunistic scheduling algorithms can make advantage of renewable energy availability to perform jobs with low priorities, at the cost of virtual machine migrations and suspend/resume functions. Another possible method for improving the effective utilization of intermittent and fluctuating renewable energy consists in using batteries to store green production surplus, and to use it during low production periods [GKL+13]. Typically for solar sources, energy can be stored during the day – if not fully consumed – and be utilized during nights when there is no production. However, batteries have an inherent energy efficiency (their yield) that leads to energy losses. The work presented hereafter has been published in:

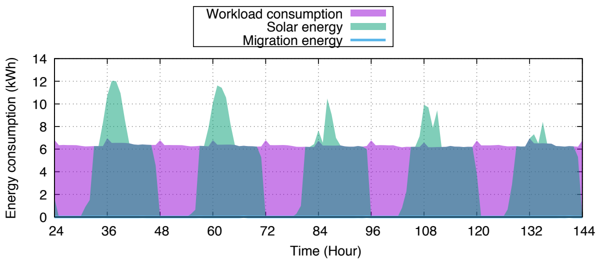
"**Balancing the use of batteries and opportunistic scheduling policies for maximizing renewable energy consumption in a Cloud data center**", Yunbo Li, Anne-Cécile Orgerie and Jean-Marc Menaud, *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, Saint Petersburg, Russia, pages 408-415, March 2017.

In this work, we discuss the main two approaches for maximizing the utilization of renewable energy in small and medium data centers, namely opportunistic scheduling and batteries. We compare these two solutions in terms of renewable energy utilization and total energy consumption in order to estimate whether the losses due to the battery efficiency balances or not the losses due to migration costs incurred by opportunistic scheduling policies. We also evaluate an intermediate solution mixing both approaches. The original paper also investigates two types of batteries (lead-acid and lithium-ion), the optimal size of photovoltaic panels and several sunlight profiles. All the simulations are done using the same scenario and traces as in Section IV.B.1.

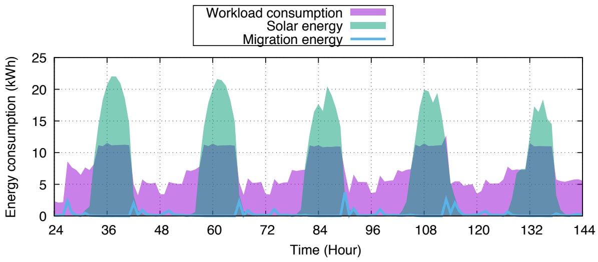
Figure IV.3 illustrates four different cases using the same battery size and the same solar panel dimension. The first case shows the baseline without ESD (energy storage device), this case leads to a wastage of solar energy. The second case shows the baseline with ESD, and in this case, solar energy can be partially stored. However, part of the solar energy is still wasted due to the limited battery size and its charging rate. The third case presents PIKA without battery, so solar energy is partially consumed by opportunistic scheduling, but the surplus solar energy is wasted. The fourth case exhibits PIKA with batteries, and it consumes almost all the available solar energy.



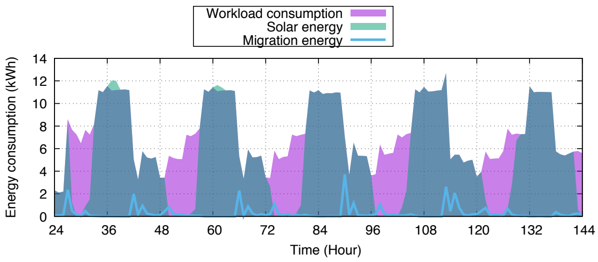
(a) Baseline without batteries



(b) Baseline with batteries



(c) PIKA without batteries



(d) PIKA with batteries

Figure IV.3 – Energy consumption of a data center with 55 servers with solar panels (160 m^2) and 40 kWh LI battery

Table IV.2 summarizes the results from the experiments displayed on Figure IV.3. It shows that PIKA-ESD approach is the most energy-efficient among all approaches. Particularity, we found that the brown energy consumption of Baseline-ESD was approximately equal to original PIKA (i.e., no ESD).

Policy	Total Energy (Wh)	Brown Energy (Wh)	Green Energy (Wh)
Baseline	768,724	442,085	326,639
Baseline + ESD	792,155	280,441	511,714
PIKA	892,458	378,569	513,889
PIKA + ESD	914,944	209,935	705,009

Table IV.2 – The energy consumption with a 160 m^2 solar farm and 40 kWh LI battery

For the opportunistic approach, the energy loss mainly depends on migrations caused by consolidation and the opportunistic scheduling delaying jobs. Since the solar energy is not sufficient enough for the workload needs, the opportunistic algorithm has to suspend some batch jobs and to perform consolidation in order to keep a low number of powered-on servers. And the delayed batch jobs then are executed when solar energy become available again. The delayed workload directly consumes the solar energy and the remaining solar energy is stored in the battery. Thus, the opportunistic approach stores less energy than baseline in the battery, and consequently, the losses due to battery efficiency are lower with the opportunistic approach. However, the total solar energy is not sufficient for the entire workload in this case, the opportunistic approach periodically performs VM consolidations that may lead to a great number of VM migrations. This migration energy cost compensates the gain.

For this reason, in the original paper, we studied solutions to partially delay the batch jobs (respectively 10, 30, 50 and 70% are delayed). In fact, when we delay less batch jobs, it leads to less migrations by consolidation, but more energy will be stored in the batteries. There is a balance for the opportunistic approach between the energy loss caused by migrations and by battery efficiency. In our case, we observed that the least energy losses are reached when 30% of batch jobs are delayed and the battery size is greater than 40 kWh.

Integrating renewable energy into data centers significantly reduces the traditional energy consumption and carbon footprint of these energy-hungry infrastructures. As renewable energy is intermittent and fluctuates with time, it is usually under-utilized. In this work, we address the problem of improving the utilization of renewable energy for a single data center by using two approaches: opportunistic scheduling and energy storage. We found an optimal solution combining both approaches that balances the energy losses due to different causes such as battery efficiency and VM migrations due to consolidation algorithms. This solution should be particularly suitable in the context of edge Clouds, since they rely on small-size data centers.

IV.B.3 Edge Cloud or core Cloud data center

Among the many challenges raised by the expanding Internet of Things (IoT), one is currently getting particular attention: making computing resources easily accessible from the connected objects to process the huge amount of data streaming out of them, as detailed in Section II.C.3. Cloud computing has been historically used as enabler for a wide number of applications. It can naturally offer distributed sensory data collection, global resource and data sharing, remote and real-time data access, elastic resource provisioning and scaling, and pay-as-you-go pricing models [AHGR14]. The work presented hereafter has been done in collaboration with Rutgers University (USA), and published in:

> “**Leveraging Renewable Energy in Edge Clouds for Data Stream Analysis in IoT**”, Yunbo Li, Anne-Cécile Orgerie, Ivan Rodero, Manish Parashar and Jean-Marc Menaud, *CCGrid: IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, Madrid, Spain, pages 186-195, May 2017.

While computation offloading to the edge can be beneficial from a Quality of Service (QoS) point of view, from an energy perspective, it is relying on less energy-efficient resources than centralized Cloud data centers [VWB+16]. On the other hand, with the increasing number of applications moving on to the cloud, it may become untenable to meet the increasing energy demands which is already reaching worrying levels. Edge nodes could help to alleviate slightly this energy consumption as they could offload data centers from their overwhelming power load [VWB+16] and reduce data movement, as detailed in Section III.C.2. In particular, as edge cloud infrastructures are smaller in size than centralized data center, they can make a better use of renewable energy [GKL+13].

In this work, we propose to leverage on-site renewable energy production in the different edge cloud nodes to reduce the carbon footprint of Cloud infrastructures incurred by IoT. Our aim is to evaluate, on a concrete use-case, the benefits of edge computing regarding renewable energy consumption. Based on the work described in Section IV.B.2, we propose an analytic model for deciding whether to offload computation from the objects to the edge or to the core Cloud, depending on the renewable energy availability and the desired application QoS, in particular trading-off between performance (response time) and reliability (service accuracy). Our validation use-case targets the Internet of Vehicles (IoV) which can be seen as a convergence of the mobile internet and the IoT [YWL+14]. In particular, we focus on video streams from cameras that need to be analyzed usually for object detection and tracking. In this particular case, as it is often the case with IoT applications, a high QoS level is required. Indeed, data lose their value when they cannot be analyzed fast enough.

Our simulations show that offloading the data to process analysis at edge significantly reduces the response time and avoids unnecessary data transmission between edge and core. Building self-producing electricity edge data centers can further reduce the traditional energy consumption and carbon footprint of Cloud infrastructures. Although this study focuses on a camera-based stream processing application, it can be applied to any other scenario where the data streams need to be processed in real-time since it provides the analytic framework for such applications.

To make advantage of on-site renewable energy facilities, Cloud managers should match their energy consumption with their renewable energy production. The intermittent nature of current most commonly-used renewable supplies (i.e. sun, wind) causes major challenges. On one hand, the renewable production can be adjusted through the use of energy storage devices. Yet, this solution remains costly and far from ideal as these devices present charge and discharge maximal rates, depth of discharge lower bounds and strong aging effects [GFKR15]. On the other hand, Cloud providers can try to adjust the workload to the energy production. For instance, opportunistic scheduling aims at postponing Cloud's workload during low-production periods to wait for renewable availability. Yet, this solution requires a portion of flexible workload (that can be postponed without impacting customers). A third option exists for distributed Clouds with several locations: geographic renewable-aware load balancing.

IV.C Distributed clouds with renewable energy

As sun and wind provide renewable sources of energy whose capacity fluctuates over time and depends on the location, distributed cloud infrastructures can benefit from several on-site productions located on distant data centers. To take advantage of such situations, follow-the-sun and follow-the-wind approaches have been proposed. The rationale is to place running VMs on resources using renewable energy, and migrate them as renewable energy becomes available on resources in other locations. However, the migration cost, in terms of both energy and performance, may be prohibitive, especially over high-latency network links. This work was done during the post-doc of Benjamin Camus within the COSMIC project (Inria exploratory action on Coordinated Optimization of SMart grIds and Clouds, 2016 - 2018) that I led.

IV.C.1 Renewable-aware scheduling for distributed data centers

The work presented hereafter has been published in:

"**A stochastic approach for optimizing green energy consumption in distributed clouds**", Benjamin Camus, Fanny Dufossé and Anne-Cécile Orgerie, *SmartGreens: International Conference on Smart Cities and Green ICT Systems*, Porto, Portugal, pages 47-58, April 2017.

In this work, we consider a distributed Cloud infrastructure comprising several data centers geographically distributed, and powered by the regular electrical grid and on-site photovoltaic panels (PV). The user management of the Cloud is assumed to be centralized as shown on Figure IV.4. Incoming users requests can arrive at any time. Each request requires to be computed by a dedicated virtual machine (VM) that can be allocated on any of the data centers. Each data center holds a given amount of homogeneous servers and they are switched off when they do not host any VM. The data centers are connected among them with dedicated wired networks.

Each data center produces its own green energy thanks to photovoltaic panels. The green energy production is not known in advance as it strongly depends on the meteorological context of each data center. When the local green energy production of a data center is not sufficient, the traditional, regular electrical grid is in charge of powering the Cloud. Following the worst case scenario, all the supply coming from the regular grid is considered as brown energy.

Our solution is named SAGITTA: a Stochastic Approach for Green consumption In disTributed daTA centers. At each time slot (i.e. each five minutes), the SAGITTA controller performs management operations. In particular, it computes the expected green power production for the next time slot as shown on Figure IV.5. The estimated value uses a reference green power production trajectory (the recorded day with more produced energy) scaled according to the last green power production. Additionally, we use a stochastic approach for estimating renewable energy production, and a greedy heuristic for allocating resources to the incoming user requests.

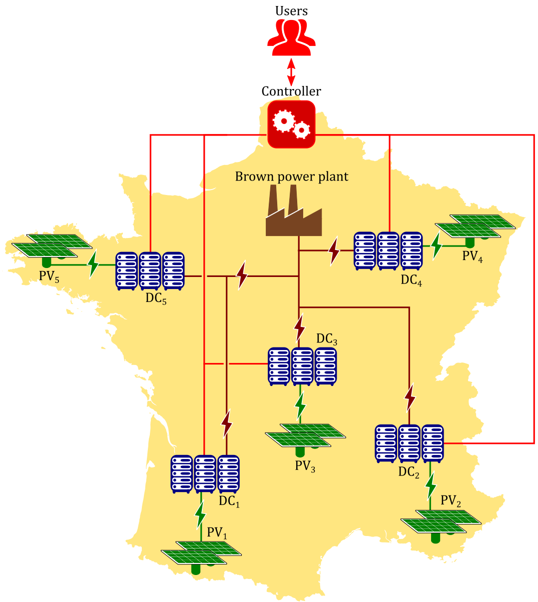


Figure IV.4 – Considered cloud model

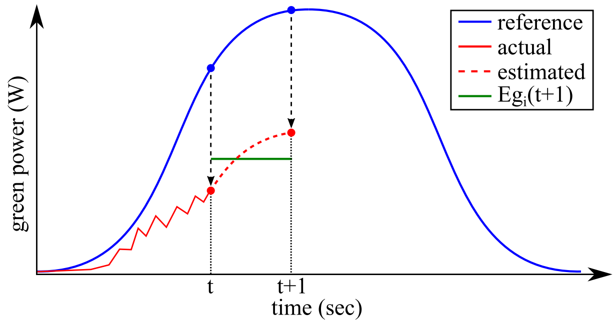


Figure IV.5 – Expected green power production computation for a time slot from t to $t+1$.

We compare SAGITTA performance against two Round-Robin inspired algorithms:

- **Round-Robin-VM** distributes the VMs fairly between the data centers regardless their green power production.
- **Round-Robin-DC** starts filling with VMs the first data center (in an arbitrary predefined order). If this data center becomes full, the algorithm starts using the next one, and so on.

Like SAGITTA, these two algorithms employ in priority the nodes already powered on.

As the performance of Round-Robin-DC strongly depends on the order of the data centers, we test two opposite configurations corresponding to the best and the worst possible contexts. To define these contexts, we sort the photovoltaic traces according to the total amount of green energy they provide. We then assign the traces to the data centers following this order. The best context corresponds to the case where the photovoltaic traces are sorted in a decreasing order. Thus, the first data center (i.e. the one filled in priority) is supplied by the best photovoltaic power trajectory.

To properly evaluate the performance of the three algorithms, it is important to note that the green power available is not always sufficient to supply the cloud needs in our simulation. That is why we also compute the ideal cumulative brown energy consumption which corresponds to the best performance reachable regarding our cloud configuration.

We simulate the cloud dynamics over one week. In this first set of experiments, we do not integrate the power costs of switching ON/OFF the servers in order to have a fair comparison to the ideal unreachable case (given by $P_B(t)$) that does not take into account these costs. Our simulation estimates that this cloud consumes a total of 4.96 MWh over the simulated week. Figure IV.6 shows the cumulative brown energy consumption of the cloud over time for the previously described scheduling algorithms. SAGITTA presents a consumption 4% above the ideal, and significantly better than Round-Robin-VM (28.8% above the ideal) and Round-Robin-DC (14.4% above the ideal in the best case, and 69.6% in the worst one).

We conduct a simulation-based evaluation using real workload traces (a normalized ClarkNet HTTP trace of [TKBL12]), servers' energy profiles measured on Grid'5000 and real production traces from photovoltaic panels (thanks to the Photovolta project [Pho]). We perform a set of experiments to determine the influence of green energy production on SAGITTA performance. As shown in Figure IV.7, the number of photovoltaic panels (PV) varies per data center and the total brown power consumption is recorded over one week. We can see that, as soon as green energy is available, SAGITTA consumes clearly less brown energy than the other approaches. We also study the performance of SAGITTA when considering switching on/off energy costs. In this case,

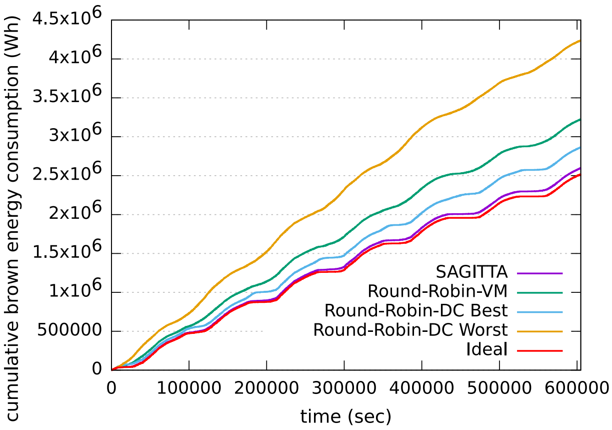


Figure IV.6 – Cumulative brown energy consumption of the cloud generated by the different approaches

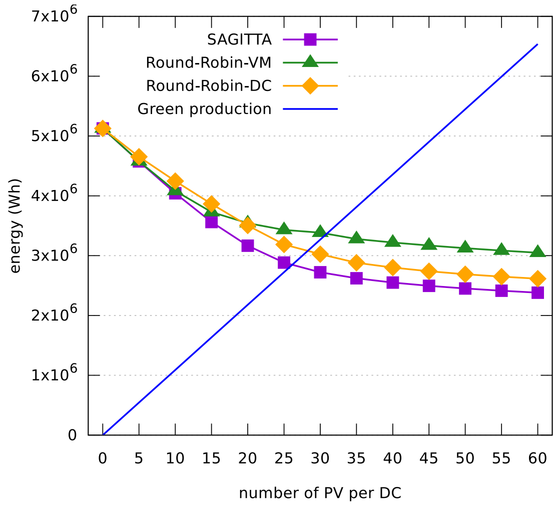


Figure IV.7 – Influence of green energy production on brown energy consumption

SAGITTA consumes 10% more brown energy than the theoretical lower bound, which is the ideal allocation, i.e. not taking into account the switching on/off energy costs.

Figure IV.7 also shows that up to about 25 photovoltaic panels, the brown energy consumption curves have a steeper slope, leading to higher gains per photovoltaic panels. For more than 25 photovoltaic panels, the energy gains are lower per added panel. When reaching 45 panels, the green energy production exceeds the total energy consumption of the data center (represented by the case with 0 panel). However, this production is concentrated during the day, whereas the workload, and consequently the energy consumption, spans over the day and the night. Thus, when reaching a number of photovoltaic panels whose production covers most of the Cloud energy consumption during daylight, adding panels can only save the energy consumption peaks at the beginning and the end of the day (when panels produce less energy), and their buying cost can thus exceed the monetary gains they generate.

The original paper also provides simulations when increasing the number of data centers (up to 40). The results show that SAGITTA can smoothly scale with the number of data centers belonging to the cloud and still outperforms round-robin strategies. These first results show that, despite VM migration energy costs, geographical load balancing can save energy for distributed Clouds with on-site renewable energy sources.

IV.C.2 Finding the optimal scheduling for green distributed data centers

Moving forward with follow-the-sun approaches, we continued our quest in search of the optimal solution to allocate VMs in a distributed cloud with on-site photovoltaic panels. Without batteries, renewable energy has to be consumed upon production or it is wasted. In this context, optimizing renewable energy consumption requires to know the local availability for the distributed cloud infrastructure, in order to adequately allocate computing resources to incoming user requests. The goal is to geographically distribute the workload among the data centers so that, it fits at best the on-site renewable energy production that is variable, not known and distributed. The work presented hereafter has been published in:

> **"The SAGITTA approach for optimizing solar energy consumption in distributed clouds with stochastic modeling"**, Benjamin Camus, Fanny Dufossé, and Anne-Cécile Orgerie, *chapter in Smart Cities, Green Technologies, and Intelligent Transport Systems*, pages 52-76, Springer (ISBN 978-3-030-02906-7), December 2018.

As in the previous section (Section IV.C.1), we consider the problem of scheduling workload across multiple data centers for minimizing renewable energy loss. We proposed an optimal algorithm based on dynamic programming to solve the problem and compare it to SAGITTA. This algorithm explores the possible configurations at a given time slot (i.e. for each data center, its number of powered on server at a given time slot) and recursively computes the energy consumption of each of them. In practice, this algorithm cannot be used to allocate VMs, as it is based on complete data knowledge and explores all the possibilities, thus inducing a combinatorial explosion.

We compare SAGITTA with the optimal solution when considering ON/OFF switches penalties. In order to compute the optimal solution, we implement the algorithm in Python. Due to the high level of computing resources required, we parallelize it and the computations are distributed for the first *forall* loop of the algorithm (concerning all the possible configurations at a given time slot). For each time slot, we use 30 servers of the Grid'5000 platform to run in parallel the optimal algorithm. Even with this optimization, we only were able to compute the optimal solution for a cloud composed of 5 data centers of 20 servers each. The algorithm took about 2 weeks (on 30 servers) to perform 1 week of simulation. The results are shown in Table and IV.3. SAGITTA is close to the optimal solution although it requires way lesser computing resources than the optimal algorithm.

Table IV.3 – Percentage of cumulative energy consumption over the optimal when considering ON/OFF penalties.

	SAGITTA	**Round-Robin-VM**	**Round-Robin-DC**
Best	5.2%	26.6%	12.9%
Worst	5.2%	26.6%	67.4%

The original paper also contains a proof of local optimality of SAGITTA and a study on the exactness of our green power production forecast. Although our solution SAGITTA provides results close to the optimal, the underlying scenario assumptions are strong:

- servers are considered homogeneous over the different data centers, in terms of computing capabilities and energy consumption;
- VMs are identical in terms of allocated resources and duration that is assumed to be one time slot;

The last assumption allows to treat VMs indifferently from each other, thus simplifying the VM allocation. Some web services hosted in Clouds can use a large number of identical VMs to treat users' requests, and with a dedicated data storage, stateless VMs can be killed and restarted upon requests arrival. This step provides strong algorithmic guarantees concerning the quality of our proposed solution. Yet, the addressed problem does not quite correspond to common Cloud use-cases. In the following, we relax these assumptions at the cost of the algorithm optimality, that we are not able to compute anymore. Yet, we ground our proposition on the same algorithms as SAGITTA, especially the allocation policy and the green power production forecast.

IV.C.3 Network-aware scheduling for green distributed data centers

On-site renewable energy production and geographical energy-aware VM allocation can be associated to lower the brown (i.e. not renewable) energy consumption of data centers. While follow-the-sun and consolidation techniques can save energy in distributed Cloud infrastructures, existing frameworks often do not consider network constraints [RLK14]. Indeed, as Cloud traffic demands diversify, network resources, inside and in-between the data centers, are often stretched to their limits and, in many cases, become a performance bottleneck [LS17]. If not carefully taken into account, network can be a major issue for energy-efficient resource management techniques, making them unfeasible in practice. The work presented hereafter has been published in:

"**Network-aware energy-efficient virtual machine management in distributed Cloud infrastructures with on-site photovoltaic production**", Benjamin Camus, Fanny Dufossé, Anne Blavette, Martin Quinson and Anne-Cécile Orgerie, *SBAC-PAD: International Symposium on Computer Architecture and High Performance Computing*, Lyon, France, pages 85-92, September 2018.

In this work, we assume that the user management of the Cloud is centralized. Also, at each time slot of 5 minutes, the Cloud receives heterogeneous VM requests in terms of memory, CPU and execution time. Their duration is known when the request is submitted. The Cloud manager is free to locate a VM on any server with enough resources to run the VM (i.e. no over-commitment). The VM location can be changed at runtime by using live migrations. We assume that servers are homogeneous for all the data centers.

We propose NEMESIS: a Network-aware Energy-efficient Management framework for distributEd cloudS Infrastructures with on-Site photovoltaic production. The originality of NEMESIS lies in its combination of a greedy VM allocation algorithm, a network-aware live-migration algorithm, a dichotomous consolidation algorithm and a stochastic model of the renewable energy supply (same as SAGITTA in Section IV.C.1) in order to optimize both green and brown energy consumption of a distributed cloud infrastructure with on-site renewable production. Our solution employs a centralized resource manager to schedule VM migrations in a network-aware and energy-efficient way, and consolidation techniques distributed in each data center to optimize the Cloud's overall energy consumption.

Due to bandwidth constraints, a data center can only migrate VMs one by one. Thus, the amount of VMs it can send in a single time slot is bounded by the sum of migration times. For a reallocation phase, our algorithm first lists the VMs to migrate for each data center. It orders the VMs by decreasing remaining execution time for each data center. The first VMs are added to the list until the sum of migration times reaches the duration of the time slot. As migration are done one by one, there is no need to consider more VMs for migration. The idea is to have the data centers with lower ERGE (expected remaining green energy) that send VMs to data centers with higher ERGE. For given receiving and sending data centers, VMs are ordered by decreasing volume (computed as the product of the VM size and its duration), and allocated one by one if some servers can host them, if the expected brown power consumption is reduced, and if the migration time constraints are fulfilled, as illustrated in the example of Figure IV.8.

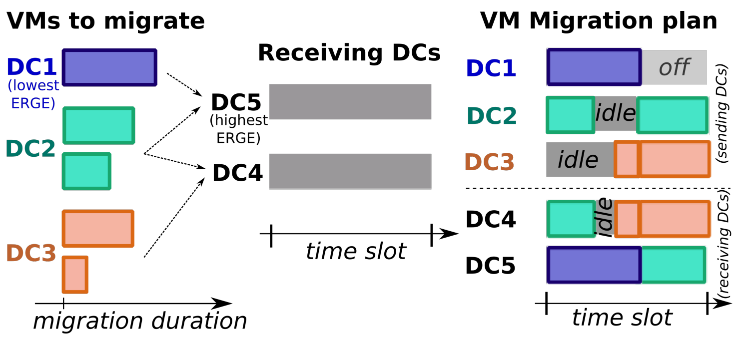


Figure IV.8 – Example of VM migrations with 5 data centers

We evaluate NEMESIS using SimGrid with real-world Cloud traces (from Eucalyptus IaaS [WB14]) and PV traces (from Photovolta [Pho]). In our experiments, we study a Cloud infrastructure with characteristics equivalent to Grid'5000 [BCAC+13] (at the time of the experiments). Overall, our Cloud comprises 1,035 servers spread across 9 geographically distributed data centers. They are linked together thanks to 10 Gigabit IP links. Inside each data center, the servers use 1 Gigabit Ethernet links.

We compare NEMESIS performance against four approaches of the literature:

- **Round-Robin** distributes the VMs fairly among the data centers regardless of their green power production;
- **First-Fit** deploys each VM on the first (according to an arbitrary predefined order) data center which can host it.
- **Modified Best Fit Decreasing (MBFD)** [BAB12] is a highly-cited approach for reducing power consumption in Clouds. It relies on a decreasing best-fit algorithm to allocate incoming VMs and to perform consolidation of running VMs. The consolidation consists in performing live migrations of VMs that run on underused servers (i.e. servers that have less than 50% of CPU used) and shutting down these servers to save energy. On the contrary to NEMESIS, MBFD does not take into account the network constraints and the local green power productions. The remaining execution time of the VMs and the energy cost of the live migration are also not considered when migrating a VM.
- **OOD-MARE** [KTB17] is another approach from literature consisting in allocating the incoming and running VMs according to the current local green power productions. With this approach, a Most Available Renewable Energy (MARE) algorithm deploys the VMs on the data center that has the highest amount of available green energy. According to an Optimal Online Deterministic (OOD) policy, the running VMs that start consuming brown energy are sequentially re-allocated using live migrations. On the contrary to NEMESIS, OOD-MARE does not rely on green production forecasts and does not perform intra-data center consolidation. It also does not consider the VM remaining execution time when performing live migration. Finally, the energy consumption of these live migrations is neglected.

The first two approaches, round-robin and first-fit, have been selected based on the approaches implemented in practice in current IaaS software stacks. Indeed, by default, Eucalyptus and CloudStack use a first fit VM placement algorithm [GRS12, Clo19], while OpenStack's default scheduler employs a combination of filters and weights to spread VMs across all servers evenly [Ope19], thus obtaining an allocation similar to round-robin with homogeneous servers.

Table IV.4 – Total cumulative energy consumption in the best/worst contexts (if different). The differences with NEMESIS are in parenthesis.

Approaches		Overall consumption	Brown Consumption
NEMESIS		17.6 MWh	11.6 MWh
OOD-MARE		17.9 MWh (1.6%)	12 MWh (3.2%)
MBFD	**best**	18 MWh (1.9%)	12.2 MWh (4.47%)
	worst		13.3 MWh (13.9%)
First-Fit	**best**	17.7 MWh (0.7%)	12 MWh (3.4%)
	worst		13.2 MWh (13.3%)
Round-Robin		17.8 MWh (1%)	12.4 MWh (6.9%)

We simulate the Cloud behavior over one week. Table IV.4 shows the total cumulative energy consumption. We can see that NEMESIS consumes significantly less brown energy than the other approaches: at least 3.18%, and 13.26% maximum. It also slightly reduces the overall (i.e. brown and green) energy consumption. During the NEMESIS execution, we observe that, as expected, it uses in priority the data center that has the highest power production. We also observe that the consolidations significantly lower the power consumption of the concerned data center. For instance in the Rennes site, a consolidation of 22 VMs occurs at time 344,100 seconds and lower the consumption of about 500 W.

Figure IV.9 shows the contributions of the three main algorithms constituting NEMESIS: Algorithm 2 concerns the migration of pre-allocated VMs, Algorithm 5 deals with inter-data center migration of running VMs and Algorithm 6 performs consolidation on each data center independently. We measure the improvement given by an algorithm by disabling it and checking the resulting increase in the total cumulative brown energy consumption. We can see that the energy

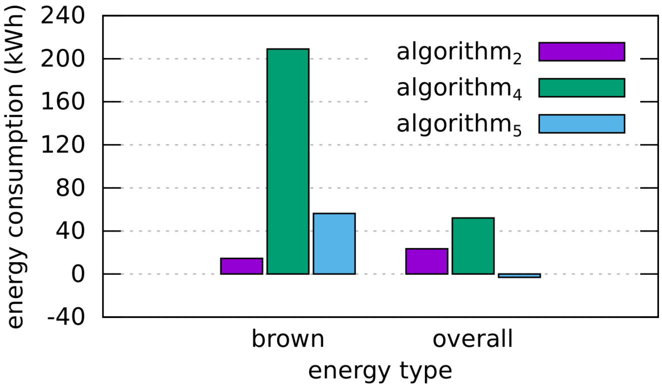


Figure IV.9 – Decrease in the energy consumption induced by the different algorithms composing NEMESIS.

consumption (both brown and global) is mainly reduced by live migrations of VMs to other data centers (i.e. Algorithm 5). The two other algorithms have only a limited positive impact on the energy consumption. This result highlights the relevance of follow-the-sun approaches in this context. Moreover, this simulation-based validation shows promising results, considering it employs a non-favorable workload for solar energy production (i.e. not exhibiting day/night patterns).

In this section, we explored on-site renewable energy production to power geographically distributed Cloud infrastructures. Distributed clouds can take advantage of multiple locations to use green energy sources with follow-the-sun approaches. Live-migration and consolidation techniques allow to further increase the savings done with renewable-aware allocation policies. Yet, these approaches rely on prediction models for both the renewable production and the Cloud workload. Furthermore, since production and workload peaks have no reason to coincide, on-site renewable facilities may be underutilized, leading to renewable energy waste. Indeed, this energy could have been injected into the electrical grid to benefit to other customers if they were needing some at this time.

IV.D Smart Grids to the rescue of energy-hungry Clouds

The growing appetite of Internet services for Cloud resources leads to a consequent increase in data center facilities worldwide. This increase directly impacts the electricity bill of Cloud providers. Indeed, electricity is currently the largest part of the operation cost of a data center [B+15]. Resource over-provisioning, energy non-proportional behavior of today's servers, and inefficient cooling systems have been identified as major contributors to the high energy consumption in data centers [GP16].

We have shown the effectiveness of using follow-the-sun approaches to lower the non-renewable energy consumption of data centers. As shown in Section IV.B.2, the renewable production can be adjusted through the use of energy storage devices. Yet, this solution presents major drawbacks due to strong aging effects on the performance of devices [GFKR15]. Instead, Cloud providers operating on-site renewable energy production, such as Apple [App17], circumvent the issue by selling their production surplus and keeping the 'green credit' of it, while still relying on the local electric grid (mostly based on coal and nuclear power) when their production is insufficient [Eps16]. Furthermore, green energy availability highly depends on the data centers' location that is fixed upon construction. Besides, Cloud data centers are required to provide a high level of availability to their customers, and consequently, some parts of the workload cannot be reshaped or postponed [HDMT18].

Smart Grids aim at dynamically and effectively adapt demand to supply to cope with the envisioned growth of both the consumers' electricity consumption and the number of distributed renewable energy sources. We believe that Smart Grids can bring flexibility in the electricity sources [GWMR17] and management [DMN+17] for distributed Clouds. For this reason, we studied their potential to share renewable energy between the Cloud sites and thus, to increase the overall self-consumption of the infrastructure.

I threw the first ideas underlying this work in a project proposal submitted in 2013 at the French research funding agency, ANR (under the young researcher program), that, repeatedly over the years, rejected it. Naming this project ICARUS (InterConnecting smArt gRids and distributed cloUds to Save energy) was certainly risky. I made, out of the short proposal version, a position paper, published in 2015 (presented hereafter in Section IV.D.1), and looked for other funding sources. The COSMIC project, presented in Section IV.C, was one of these other sources, and it provided me means to explore one of the research axes presented in the original project. My work on another of these initial axes started at the end of the COSMIC project and is presented hereafter. It consists in exploring interconnections between distributed Clouds and Smart Grids.

IV.D.1 Interconnecting distributed Clouds with Smart Grids

It is estimated that 10% of electric energy produced by power plants is lost during transmission and distribution to consumers, with 40% of these losses occurring on the distribution network [FPY+09]. As an example, in 2006 in the United States, the total energy losses and distribution losses were about 1,638 billion and 655 billion kWh, respectively [FPY+09]. To reduce these losses, more distributed energy management policies are required. To this end, Smart Grids are expected to provide the means to control the energy supply more efficiently and to dynamically manage peak load. They could also help in increasing the part of renewable energy in the electrical mix used by highly distributed Clouds, without asking for the deployment of on-site renewable energy sources owned by the Cloud provider. The work presented hereafter has been published in:

"**Interconnecting Smart Grids and Clouds to Save Energy**", Anne-Cécile Orgerie, *SmartGreens: International Conference on Smart Grids and Green IT Systems*, Lisbon, Portugal, pages 1-6, May 2015.

Advances in distributed systems have historically been related to improving their performance, scalability and quality of service. Yet, it is now urgent to drastically increase the efficiency of large-scale distributed systems in order to curb the rising energy consumption of ICT. Indeed, as detailed in Section I, ICT energy consumption now exceeds 3% of the global energy consumption, and this value is growing at a rate of 9% per year [The18]. As outlined in [Mil13], a faster growth in ICT energy use, with the information appetite of Big Data, means big networks and big infrastructure which unavoidably leads to big power.

While Clouds come naturally to the rescue of Smart Grids for dealing with their big data issue (due to numerous sensors), little attention has been paid to the benefits that Smart Grids could bring to distributed Clouds. We propose to study the opportunity for Smart Grid technologies to come to the rescue of highly-distributed energy-hungry Clouds. Unlike in traditional electrical distribution networks, where power can only be moved and scheduled in very limited ways, Smart Grids dynamically and effectively adapt supply to demand. This recent technology offers the unique chance to monitor the energy consumption in real-time of entire distributed infrastructures through their smart sensors, to reduce energy use through their smart actuators, and to act as the bridge enabling hitherto impossible joint energy production/consumption synergies.

On the other side, through adaptation mechanisms that dynamically shape distributed Cloud's workload –and consequently energy consumption– data centers could provide more flexibility to Smart Grids in their electricity management. We would like to explore this win-win strategy for collaborating between Clouds and Smart Grids where: (1) Smart Grids bring their renewable energy availability to Clouds, and (2) Clouds bring their energy consumption flexibility to Smart Grids. We explore these two directions one after the other in the following sections.

IV.D.2 Exchanging renewable energy between data centers

The work presented hereafter has been published in:

"**Self-Consumption Optimization of Renewable Energy Production in Distributed Clouds**", Benjamin Camus, Anne Blavette, Fanny Dufossé and Anne-Cécile Orgerie, *IEEE Cluster Conference*, Belfast, United Kingdom, pages 359-369, September 2018.

"**Harnessing the geographical flexibility of distributed computing clouds for cooperative self-consumption**", Benjamin Camus, Anne Blavette, Fanny Dufossé and Anne-Cécile Orgerie, *ISGT Europe: IEEE PES Innovative Smart Grid Technologies Conference Europe*, Sarajevo, Bosnia and Herzegovina, pages 1-6, October 2018.

Demand-side management is key for increasing the level of renewables in the energy mix. This concept implies modifying the consumer's energy consumption by means of different incentives (e.g. economic) with respect to the fluctuations of the renewable electricity production in order to maintain the safe and stable operation of the power system. So far, a number of flexible load types has been extensively studied for demand-side management applications, such as controllable water heaters, air-conditioning equipment, refrigerators, etc. as well as electric vehicles. Traditionally, flexible loads are characterized by a certain degree of temporal flexibility. This means that their energy consumption can be reduced partially or entirely (i.e. interruptible loads) during a certain amount of time, usually with a rebound effect, or that their activation can be postponed (i.e. deferrable loads).

However, it may also be interesting to consider spatial flexibility. Spatial flexibility can be defined as the ability of a load to migrate physically from one node in the electrical network to another in a sufficiently short amount of time to be relevant for demand-side management. Spatial flexibility may be used for addressing local electrical network issues, such as line congestion or voltage control. It is important to note that one approach to solve the mentioned local network issues consists in curtailing the renewable electricity generation, thus losing energy. Hence, spatial flexibility can represent an alternative to curtailment. Another advantage of spatial flexibility consists in accompanying the spatial fluctuations inherent to variable renewable electricity generation from wind or photovoltaic sources. In other words, such loads could be expected to migrate from one region to another of the power system if the electricity generation from renewables becomes more important in the latter than in the former. This represents a relevant manner to further harness renewables, in complementarity with temporal flexibility.

Spatial flexibility can be provided by loads such as distributed computing clouds. Indeed, as proposed in Section IV.C, VMs can migrate from a data center where renewable electricity generation is becoming scarce to another with more favorable conditions. Spatial flexibility represents an interesting approach for cloud managers, as they are becoming more and more equipped with photovoltaic (PV) panels in the perspective of self-consumption to reduce their financial costs and to increase their renewable energy part. VM migration can thus allow to increase the cloud self-consumption, while not impacting the quality of service provided to the cloud customers. Although several studies focus on VM migration for better harnessing renewables (as the one presented in Section IV.C), none has compared this approach with the alternative which consists in exchanging photovoltaic energy between the data centers through the electrical network.

In this work, we propose to rely on the flexibility brought by Smart Grids to exchange renewable energy between data centers and thus, to further increase the overall Cloud's self-consumption of the locally-produced renewable energy. Our solution is named SCORPIUS: Self-Consumption

Optimization of Renewable energy Production In distribUted cloudS. It is based on our previous proposition NEMESIS, presented in Section IV.C.3. In addition, it optimizes the Cloud's self-consumption by trading-off between VM migration and renewable energy exchange. This optimization is based on an original Smart Grid model to exchange renewable energy between distant sites, that is based on the principles of collective self-consumption. In our context, we propose an extended, multi-site version of the collective self-consumption approach, where data centers located over a wide geographical area (e.g. over a entire country) could exchange their excess of photovoltaic electricity between themselves. However, exchanging energy, even though it is considered to be performed for free between data centers, implies using the electrical network, which also comes at a cost. As current pricing systems, we propose a cost based on network use tariffs, to cover technical and non-technical sub-costs, such as:

- power losses in the distribution equipment (mostly lines and cables), for instance due to their electrical resistance,
- electrical equipment aging,
- grid management services such as metering,
- etc.

We compare the economical performance of the two mentioned approaches, namely VM migration and energy exchange, as well as the performance of an approach combining both (i.e. SCORPIUS). In order to perform these comparisons, we simulate with SimGrid [Sima] two days of Cloud execution using Google traces [RWH11], and take the results only of the second day to observe the steady state. During the simulation, we compute cumulative total and brown energy consumption of the Cloud. We also compute its local self-consumption ratio which corresponds to the ratio of the PV energy consumed locally by DCs by the total amount of energy it consumes. Finally, we compute the collective self-consumption ratio which is the ratio of PV energy consumed by the cloud, including the virtual energy exchange (in the case where our energy model is considered), by the total amount of consumed energy. On the Cloud management side, we compare three classical approaches defined in Section IV.C.3: round-robin, first-fit and MBFD (Modified Best Fit Decreasing [BAB12]). Table IV.5 shows these results with the current implementations' curves representing the behavior of Clouds stacks that do not shutdown servers (all the other curves use shutdown).

Table IV.5 – Simulated overall cumulative cloud performance.

<table>
<tr><th colspan="3"></th><th>total consumption</th><th>brown consumption</th><th>local self-consumption</th><th>collective self-consumption</th><th>green lost</th></tr>
<tr><td rowspan="3">Current implementations</td><td colspan="2">Round Robin</td><td>4.11 MWh</td><td>2.96 MWh</td><td colspan="2">27.92 %</td><td>22.18 %</td></tr>
<tr><td rowspan="2">First-Fit</td><td>best</td><td rowspan="2">4.10 MWh</td><td>2.88 MWh</td><td colspan="2">29.92 %</td><td>16.78 %</td></tr>
<tr><td>worst</td><td>3.10 MWh</td><td colspan="2">24.42 %</td><td>32.07 %</td></tr>
<tr><td rowspan="5">State-of-the-art solutions</td><td colspan="2">Round Robin</td><td>3.27 MWh</td><td>2.27 Wh</td><td colspan="2">30.47 %</td><td>32.43 %</td></tr>
<tr><td rowspan="2">First-Fit</td><td>best</td><td rowspan="2">3.25 MWh</td><td>2.14 MWh</td><td colspan="2">34.03 %</td><td>25.03 %</td></tr>
<tr><td>worst</td><td>2.54 MWh</td><td colspan="2">21.85 %</td><td>51.88 %</td></tr>
<tr><td rowspan="2">MBFD</td><td>best</td><td>3.42 MWh</td><td>2.29 MWh</td><td colspan="2">33.2 %</td><td>22.91 %</td></tr>
<tr><td>worst</td><td>3.48 MWh</td><td>2.72 MWh</td><td colspan="2">21.5 %</td><td>49.32 %</td></tr>
<tr><td rowspan="6">With energy exchange</td><td colspan="2">Round Robin</td><td>3.27 MWh</td><td>2.06 MWh</td><td>30.47 %</td><td>37.14 %</td><td>17.66 %</td></tr>
<tr><td rowspan="2">First-Fit</td><td>best</td><td rowspan="2">3.25 MWh</td><td rowspan="2">2.04 MWh</td><td>34.03 %</td><td rowspan="2">37.08 %</td><td rowspan="2">18.33 %</td></tr>
<tr><td>worst</td><td>21.85 %</td></tr>
<tr><td rowspan="2">MBFD</td><td>best</td><td>3.42 MWh</td><td>2.19 MWh</td><td>33.2 %</td><td>36.13 %</td><td>16.12 %</td></tr>
<tr><td>worst</td><td>3.48 MWh</td><td>2.23 MWh</td><td>21.51 %</td><td>35.86 %</td><td>15.49 %</td></tr>
<tr><td colspan="2">SCORPIUS</td><td>3.25 MWh</td><td>2.05 MWh</td><td>33.47 %</td><td>36.82 %</td><td>18.94 %</td></tr>
</table>

The proposed energy exchange model significantly reduces the brown consumption thanks to energy exchanges. We can observe that, thanks to this approach, the green energy losses are reduced by 13% for Round Robin and 34% (respectively 5%) for First-Fit and MBFD in the worst (respectively best) scenario. SCORPIUS exhibits the best performances among all the evaluated approaches. Indeed, its total and brown energy consumption values are lower than all the tested Round-Robin variants, with 3.27 MWh for the total consumption of Round-Robin in the best case, against 3.25 MWh for SCORPIUS, thus saving at minimum 0.02 MWh per day. Besides, when comparing SCORPIUS with the best First-Fit approach that switches off servers but does not use energy exchange (i.e. representing state-of-the-art approaches), we are able to save about 0.09 MWh per day of brown consumption on the Google workload. This 0.09 MWh of saved brown energy for SCORPIUS represents an extrapolated saving of 32.9 MWh per year for the Cloud (i.e. its brown energy consumption decreases of around 7.6%).

In all the cases, the best First-Fit consumes less brown energy than Round-Robin and has a higher self-consumption ratio. However, it is worth noting that this best scenario is very unlikely to occur in real production systems as it represents an ideal data center ordering in terms of green production. Moreover, as we can see in the worst scenario, when there is no renewable energy exchange, First-Fit can consume significantly more brown energy than Round-Robin. Only the (unlikely) best First-Fit approach achieves a brown consumption of about 0.01 MWh less than SCORPIUS. But, in this case, as in the others, SCORPIUS has the highest local self-consumption, which means that it uses the electric grid (and the virtual pool) less than the other approaches, and has therefore less impact on it.

Our approach combines energy exchanges and virtual machine migrations among data centers in order to increase the renewable energy consumption. The original papers presents other simulations on various Cloud production traces to show the effectiveness of SCORPIUS in comparison with currently implemented methods and state-of-the-art solutions. Note that the second paper presented in this section has been published in an electrical engineering conference (managed by the IEEE Power & Energy Society). As we softly shift towards electrical grid concerns, moving VMs between data centers can be seen as a demand-response strategy where the Cloud workload is shed on a given location to adapt the local production at a given time.

IV.D.3 Managing Smart Grids

Smart Grids embed entire distributed computing infrastructures to manage electrical networks. These infrastructures could also benefit from energy-efficient techniques in order to improve their design and management. Yet, at first, it is necessary to analyze the tight interdependence between this monitoring infrastructure and the working of the electrical network itself. This work has been initiated in the context of the RennesGrid project (Ademe project led by Schneider Electrics, 2017 - 2020). Our aim in this project is to evaluate the impact of the ICT network on the electrical grid that it monitors. The work presented hereafter has been published in:

"**Co-simulation of an electrical distribution network and its supervision communication network**", Benjamin Camus, Anne Blavette, Anne-Cécile Orgerie and Jean-Baptiste Blanc-Rouchossé, *IEEE Consumer Communications & Networking Conference*, Las Vegas, United States, January 2020.

In this work, we study a concrete usecase based on the publicly available "European Low Voltage Test Feeder" electrical network model [IEE19]. It provides power consumption time series for each electrical load it contains. They represent the consumption of 55 electrical loads in the same district, each seemingly representing a single household. The electrical grid between the homes exhibits a tree topology that is connected to the district power substation through an electrical line named *Line1* in the following.

An arbitrarily-chosen number of 15 electric heaters (direct-acting) were added to this electrical network. The power consumption of each of these heaters was modeled as a cyclic profile alternating between a typical value of 2 kW and 0 kW. The power profile of a single radiator, observed as part of an experiment, was used to model the power consumption profile of all the radiators modeled in this article, to which random time delays were applied to model the aggregation effect. It must be noted that, in the absence of additional experimental data, a simple heater model was considered. In this model, the post-shedding rebound effect on the power consumption is not included. However, in the case where the shedding duration is relatively short, it may be assumed that the corresponding rebound effect is also small.

Line congestion occurs when the current flowing through a line exceeds its rated value, and it should be avoided. In the past, this type of issue used to be prevented by a sufficient over-sizing of the electric network. However, grid upgrading is extremely costly and time-consuming. Hence, distribution system operators (DSOs) seek now to maximize their assets usage by deploying smart energy management strategies. Short-term load shedding may be one of them. This strategy consists in suspending temporarily the power supply to given electric loads. In particular, the shedding of heaters over a sufficiently short period of time may have a negligible influence on the consumer's thermal comfort. However, the successive and repeated shedding (called "cascado-cyclic shedding" [VPC+15]) of a sufficiently important number of radiators may solve a line congestion issue. It is important to note that automated shedding, as opposed to consumer-activated shedding, is necessary to harness this flexibility potential. This strategy requires indeed a short reaction time and may potentially need to be repeated a significant number of times.

To support automated shedding in the electrical network, we consider the following TCP/IP communication network. The households and Line1 are equipped with computing devices that are controlled and monitored. All these devices are connected through Ethernet links on the same local network. For the sake of simplicity, we consider a star topology with homogeneous bandwidth and latency between the nodes, similarly to the connection of each house of the district to the district DSLAM (digital subscriber line access multiplexer) of a single Internet Service Provider.

We consider the following cascado-cyclic policy to automate the shedding of the household heaters in the electrical network. A shedding sequence is initiated when the current in Line1 reaches a given upper threshold. Several households are then selected for shedding. After a short time, a new iteration of the process starts: a new group of households is selected and the shedding process switches to these households. This process is repeated until the current in Line1 falls below a lower threshold. At this point, the shedding process is suspended. To ensure that all the households receive the same amount of shedding commands, they are selected in a cyclic way.

This policy can be implemented by several algorithms. Here, we propose two of them, that are representative of respectively centralized and decentralized approaches (see Figure IV.9(a) and IV.9(b) respectively). The decentralized approach avoids that a single actor gathers information on all the consumers at very short spatiotemporal scales. Thus, it may be considered by the electric grid customers as less intrusive. At the opposite, with the centralized approach, the DSO manages directly all the system. It conserves then a higher level of confidence regarding its own ability to operate safely and reliably its local network.

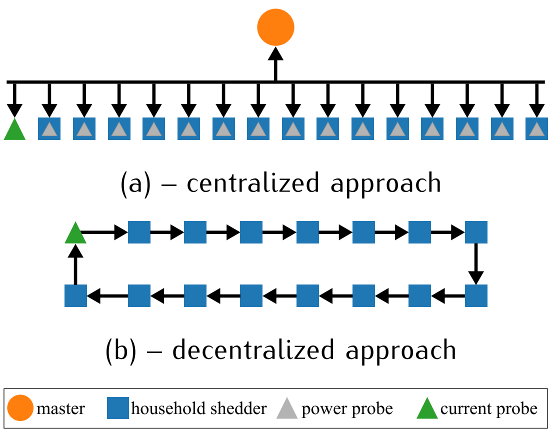


Figure IV.10 – The two implementations of the cascado-cyclic process

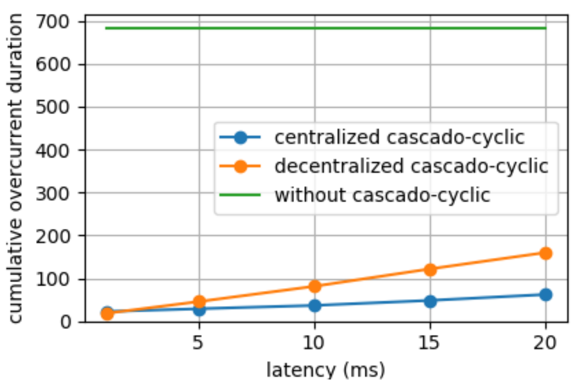


Figure IV.11 – Cumulative overcurrent duration vs. latency.

To compare the centralized and decentralized versions of the cascado-cyclic process, we model the pilot network using SimGrid, and the electric network using PowerFactory, a commonly used power system simulator [DIg]. We use the FMI++ PowerFactory FMU export utility [FMI] and our SimGrid-FMI plug-in (detailed in Section II.D.3) to co-simulate the entire smart grid. Our experiments run on a single machine. PowerFactory runs on Windows 10 whereas SimGrid is executed in Windows Subsystem for Linux. We use our FMU proxy to communicate between the two environments. During the co-simulation, SimGrid activates and deactivates household heaters shedding in PowerFactory. It reads the power and current consumption values from PowerFactory to simulate the probes behavior. We also perform a monolithic simulation with PowerFactory (i.e. without SimGrid) to observe the system trajectory when no shedding is performed. We use this trajectory as a baseline to reflect the impact of the control system on the smart grid operation, and to validate the co-simulation.

Figure IV.12 shows the evolution of the current in Line1 over time with the centralized approach and a communication network latency of 1 ms (between each node and the central point of the star network). From this graph, we see a nominal behavior of the co-simulation. The cascado-cyclic process starts when the current in Line1 reaches the upper threshold. Then, the current decreases due to the shedding of the heaters. When the current falls below the lower threshold, the cascado-cyclic process stops. Then, the trajectories of the co-simulation and the monolithic PowerFactory simulation coincide perfectly due to the absence of shedding. We get similar validity results with the decentralized version.

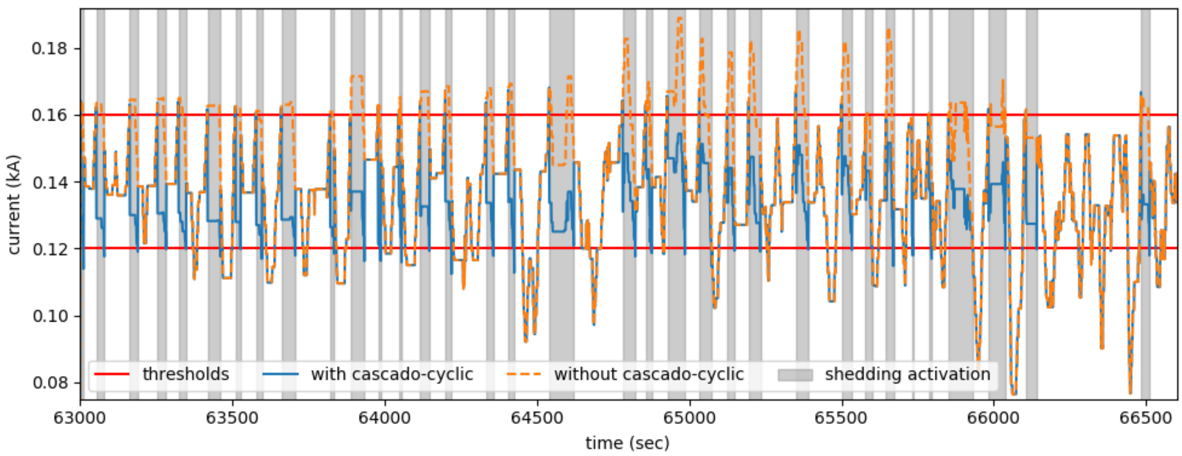


Figure IV.12 – Evolution of the current in Line1 over time with the centralized implementation.

Figure IV.11 shows that with a communication network latency of 1ms, the two approaches greatly reduce the cumulative overcurrent duration of approximately 97%. In this context, the latency is so low that the impact of the communication network becomes negligible and the two cascado-cyclic implementations offer similar results. However, with higher latency, the decentralized implementation is more sensitive to the communication network latency than the centralized version. When the latency increases to 10ms (resp. 20ms), the cumulative over-current duration of the decentralized implementation is 2.2 (resp 2.6) times higher than with the centralized approach. Indeed, the shedding command packet may be forwarded several times from household to household before it is received by a sheddable one. At the opposite, the shedding commands are directly sent to the right households with the centralized implementation.

For this work, we implemented a co-simulation tool named SimGrid-FMI to jointly simulate electrical grids and their management communication network using PowerFactory on the electrical side (or any FMI-compliant power system simulator) and SimGrid on the communication side. This framework is open-source and available online and can be directly used by the smart grid community to obtain sound co-simulations [Simb]. Thanks to this framework, we study the influence of the communication network QoS on the management of an electrical grid. Observations based on metrics relevant to the power system community, such as cumulative shedding duration per heater, etc. are provided in the original paper. This simple example opens new research avenues for our co-simulation framework and for the study of the interactions between Smart Grids and distributed ICT systems.

The electricity consumption keeps growing worldwide: the number of consumers increases along with the individual consumption. According to the International Energy Agency, in 2018, the worldwide electricity consumption increased by 4%, the fastest pace since 2010 [Int19]. The current challenges in the energy sector are multiple: integration of increasingly flexible distributed loads, such as electric vehicles, exploitation of renewable energy sources, often variable by nature and decentralized, and energy storage management. The digitization of electricity grids is expected to meet these challenges by jointly optimizing the production, distribution and consumption.

Smart grids involve the large-scale deployment of communication means to interconnect the electrical devices and to autonomously pilot their management. This digital infrastructure relies on computing, communication and storage resources to provide secure tools for processing, modeling, predicting and optimizing the electrical grid utilization.

Current telecommunication networks experience delays, jitter and bottlenecks. Such a dynamicity, ubiquitous in Internet networks, can greatly impact the management of smart grids as they require a guaranteed quality of service [XD16, GHT17, R$^+$17]. Consequently, management solutions need to carefully take into account mutual impacts between the electrical network and its management telecommunication network.

After switching off servers (Section III.B) and network devices (Section III.C), we continued with heaters in this section, showing once again the power of switching off policies to save energy and electric cables. We also demonstrated on a simple example the potential impacts of communication network latencies on a distributed management policy for the electrical network. Now that we have the right tool to observe and simulate such effects, we plan to find solutions for optimizing the smart grid management.

IV.E Perspectives

This chapter outlines my efforts in greening ICT infrastructures. Recently, data centers started to exploit renewable energy sources to decrease their carbon emissions, in particular solar panels, wind turbines and biogas fuel cells. Cloud providers can then either distort the workload with opportunistic scheduling (as in Section IV.B.1) or smooth the renewable energy production with batteries (as in Section IV.B.2). Moreover, distributed Cloud systems can perform follow-the-sun policies in order to increase the green energy use, as proposed in Section IV.C. Follow-the-sun and consolidation techniques rely on VM migration capabilities that directly depend on telecommunication network bandwidth, inside and in-between the data centers. Yet, as explained in Section IV.C.3, these network constraints, usually underestimated in literature [RLK14], and the related energy consumption prevent VM migration from being the optimal solution that dynamically adjusts the Cloud workload and the on-site electricity generation. Along this third research axis (and last one in this manuscript), I gently drifted towards electrical engineering, and especially smart grids, with *smart* meaning, as often, *ICT-equipped*. The work presented in this chapter has been mostly performed during the PhD thesis of Yunbo Li (currently post-doc at Beihang University in Hangzhou, China) and the two consecutive post-doc positions of Benjamin Camus (currently research engineer at Scalian).

Sustainable clouds. Internet keeps growing, and the cloud systems that supply it go the same way. Fueling the clouds with renewable energy gives the hope to reduce carbon emissions due to ICT. Yet, follow-the-renewable policies for geographically distributed clouds intrinsically lean on over-sized and redundant infrastructures. The resulting carbon emissions reduction only concerns the use phase, and the repercussion could be larger on the manufacturing phase, if such policies require more hardware resources. Indeed, even in the case where all carbon emissions due to electricity consumption are compensated through the production of renewable energy, infrastructures embedding more hardware resources present a higher material footprint (i.e. they involve the utilization of more raw materials, whose quantity can be very limited depending on these materials). I plan to investigate cloud systems powered only by renewable energy and to study the indirect energy costs of such systems.

Smart systems. As I headed towards the interdisciplinary issues of Smart Grids, I started to observe, from the inside, a smart system, designed for improving an existing non-ICT system. Examples of such digitalization are numerous and various: smart buildings, smart factories, smart cities, etc. All these smart systems aim at using communication sciences and technologies to improve the performance, and targeted metrics related to their functioning. Yet, even if energy consumption or carbon footprint belong to these metrics, the impact of the ICT system itself, which is in charge of piloting the underlying system, is often not considered in the overall metrics. It looks as if these smart ICT systems are invisible, uncountable, so powerful in optimizing what they intend to optimize (i.e. energy in the case of Smart Grids), that there is no need to worry about their own energy cost. In the case of Smart Grids, I intend to quantify and to optimize the energy cost of ICT resources used to pilot the electrical grid within the context of the RI/RE project[2] that I am leading.

Ecodesign. While I have been focusing on the use phase of ICT equipment, it often does not constitute the most impacting phase of the lifecycle. Indeed, when considering the equivalent carbon dioxide (CO_2e) as the global warming metric, the production phase has a significant impact, especially for end-user devices, and even when considering virtualized resources [And13]. Ecodesign aims at reducing the ecological footprint when considering the whole lifecycle, at both hardware and software levels [VP16]. Concretely, for a cloud service, it means improving the energy-efficiency of the service, but also reducing the employed cloud resources, increasing the lifetime of hardware resources and the sobriety of software systems. And if we go even a step further, it could mean reducing quality or availability of the service for given time periods or certain applications. Current ICT carbon track record calls for new compromises between carbon emissions and quality of service. I intend to scout for such tradeoffs and methodologies in order to design sustainable ICT systems.

[2]CNRS Momentum project on Optimizing the smartness of electrical grids (2019 - 2022) `http://people.irisa.fr/Anne-Cecile.Orgerie/RI/RE/`

And we were free.
Once we took off,
we didn't give a damn of
what was going on below.
We had no contact with Earth,
there was no control tower,
there was nothing.
We were masters after God.
Absolutely.
I liked that.

Adrienne Bolland

V Conclusions and perspectives

This manuscript describes the three main challenges that I explored since October 2012:

1. understanding the energy consumption of distributed infrastructures (Chapter II);
2. improving the energy efficiency of distributed infrastructures (Chapter III);
3. greening distributed infrastructures (Chapter IV).

Some perspectives and future work have been described at the end of each chapter in Sections II.E, III.E and IV.E. This chapter concludes this manuscript and provides general perspectives.

The understanding challenge involved measurements on real infrastructures, design of models and metrics, and implementation and validation of simulation tools for distributed infrastructures. Through this step, I tackled reproducibility issues, inherent in the experiments that constitute a consequent aspect of my work. Their analysis provides a keen comprehension of the real non-idealized ICT world. It led me to explore energy models of virtual distributed systems, whose numbers of devices and software layers keep increasing. Indeed, connected devices represent only the tip of the iceberg because they heavily depend on intertwined distributed computing, storage and communication infrastructures. Such a complexity must be unraveled to identify the sources of wasted energy. Towards this end, I will pursue my efforts in the implementation of end-to-end simulation tools able to express clear and comprehensive trade-offs between quality of service and energy consumption of distributed infrastructures. I hope these tools will eventually be useful for diagnosing the energy consumption of operational infrastructures, and for exploring what-if scenarios in order to guide the design of such infrastructures.

The improving challenge covered the non-proportionality of ICT resources, the design of Cloud infrastructures and the end-users' involvement in energy savings. This last point is the key to curve down the ICT energy consumption. Indeed, as displayed on Figure I.1, the global number of connected devices grows faster than the number of Internet users. Although, globally, between 2012 and 2019, the number of IP devices per Internet user slightly diminished from 5.57 to 5.12 on average (−8%), the energy consumption per Internet user slightly heightened (+4%), as well as the energy consumption per IP device (+12%). Meanwhile, the number of Internet users globally went from 34% of the population in 2012 to almost 56% in 2019. Yet, these average numbers hide significant heterogeneity among the population in terms of Internet devices, and plead for a better resources distribution, if we want to globally reduce ICT's environmental impact. Towards this end, engaging users through efficient and adequate means is crucial in order to offset the current snowballing trend while avoiding rebound effects.

The greening challenge implied first a single data center partially powered by on-site solar energy, then distributed clouds with renewable energy sources, and finally smart grids to the rescue of energy-hungry Clouds. Exploiting renewable energy sources comes with temporal and spatial constraints that are currently hidden by electrical grids. Yet, the increase of renewable share in the electricity mix will stress the challenge of efficiently distributing this utility. In this context, ICT can be seen as a flexible consumption charge and an optimization means. This path led me to interdisciplinary projects at the intersection of electrical engineering and computer science issues. Such interdisciplinary challenges flourish thanks to smart systems that leverage computational sciences for optimization purposes. Yet, the efficiency and sustainability of such smart systems often remains to be proved. The electrical grid is an indispensable example of these smart systems and its optimization promises great advances in terms of efficiency and integration of renewable sources.

In practice, the explorations described in this manuscript were more entwined than the way they are presented here for clarity's sake. The actual order has been complicated by the chronology of research projects, PhD theses and post-doctoral supervisions, even if they firstly were chances of fruitful and interesting collaborations. Experiments took a large part of these efforts and always required many more runs than initially planned. Although harder to publish than novel algorithms and methods, experimental measurement analyses provided solid contributions to build sound models and to develop reasonable hypotheses. They gave me the satisfaction of understanding a bit more how ICT devices and systems really work despite their intrinsic complexity and layers intricacy. Furthermore, this experimentation taste led me to refine the methodological aspects of estimating the energy consumption of ICT systems, and especially virtualized systems. Such a knowledge is valuable to evaluate the upcoming technologies, like 5G for instance, the 5th generation of digital cellular networks. Indeed and unfortunately, in the very competitive sector of ICT, new technologies often start to be deployed before any analysis of their impact.

The results presented here were paved by real hardware experiments and also true human collaborations with colleagues who provided invaluable ideas, hands, coffees, and many more indispensable contributions. As it is my first (and hopefully last (forever)) HDR manuscript, I wanted to insist on these collaborations by citing the papers as tangible proofs that I was not alone on these explorations. These numerous interactions and collaborations with my colleagues since 2012 (and before) were enjoyable, and essential in shaping the propositions and achieving the results presented in this manuscript which I had to write alone though. The complete list of my 130 co-authors from October 2012 to January 2020 can be found at the end of this manuscript.

Stretching the limits of current and future ICT infrastructures constitutes more than an electrifying challenge: this could drive computer scientists to the design of robust systems without heavily relying on redundancy and over-provisioning. Indeed, since several years now, distributed systems offer high availability and fault tolerance at the cost of large energy expenses as they routinely execute tools, like the Netflix Chaos Monkey [IT12] for instance, to randomly inject software and hardware failures. The frequent replacement, and consequently short lifetime, of Cloud servers for reliability purposes constitutes another example of the energy cost of distributed systems' guarantees. Such mechanisms ground quality of service on redundancy and over-provisioning although these techniques necessitate additional devices and increased power consumption. Their business models often lean on advertisement-dependent *free* services and volume-unlimited subscriptions that drive user consumption. Low power infrastructures could be designed by inventing new QoS compromises between end-users on one side, and resource and application providers on the other side. The wide adoption of such compromises will depend on their careful design, easy management, and incentive economical model.

Along with ICT-related consumption, global energy consumption continues to grow, weighing more and more heavily on global greenhouse gas emissions. To overcome this problem, the energy sector seeks to reduce its dependence on fossil fuels. However, renewable energy sources, which are often intermittent and distributed by nature, require controlled integration into the electricity grid in order not to compromise the essential balance between production and consumption. Smart grids advocate for a paradigm shift: the move from a passive distribution grid to a multi-stakeholder distributed grid able to deal with flexible loads, variable sources and distributed storage. This shift currently happens through the digitization of the energy sector, and consequently, in practice, the massive and large-scale deployment of interconnected ICT systems to manage the grid. This means: computing, communication and storage resources to provide secure digital data processing tools, dynamic system modeling, load prediction and optimization of electricity management. On the way to a more sustainable world, the smart digitization of electrical grids requires optimized ICT infrastructures for managing the electrical grids – infrastructures that guarantee the grids' performance while minimizing their own energy consumption. This is the goal of the work that I recently started.

As societal and environmental challenges become more and more indisputable, I am convinced that scientists can help in building a sustainable future. Computational sustainability aims at providing methods based on computational tools for a sustainable environment, economy and society [G+19]. While human well-being and protection of the Earth are at stake, computational sciences should be employed with care: they also belong to the problem as they consume consequent amounts of resources and energy. Ideally, the deployment of such tools at large-scale would be conditional to an environmental impact assessment that would analyze the improvements they bring in the light of their overall costs. I already highlighted the complexity of conceiving and validating comprehensive models, even when considering only the electricity consumption during the use phase. Taking into account the whole life cycle and more metrics further complicates this issue. Yet, we need such holistic models to get the chance to propose energy-efficient tools for computational sustainability.

References

[AFGM$^+$15] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys Tutorials*, 17(4):2347–2376, 2015.

[AHGR14] S. Abdelwahab, B. Hamdaoui, M. Guizani, and A. Rayes. Enabling Smart Cloud Services Through Remote Sensing: An Internet of Everything Enabler. *IEEE Internet of Things Journal*, 1(3):276–288, June 2014.

[All87] David W. Allan. Time and Frequency (Time-Domain) Characterization, Estimation, and Prediction of Precision Clocks and Oscillators. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 34(6), November 1987.

[And13] A. S. G. Andrae. Comparative Micro Life Cycle Assessment of Physical and Virtual Desktops in a Cloud Computing Network with Consequential, Efficiency, and Rebound Considerations. *Journal of Green Engineering*, 3:193–218, 2013.

[App17] Apple. Environmental Responsibility Report – 2017 Progress Report, Covering Fiscal Year 2016. Technical report, Apple Inc., April 2017.

[AS17] A. Al-Shuwaili and O. Simeone. Energy-Efficient Resource Allocation for Mobile Edge Computing-Based Augmented Reality Applications. *IEEE Wireless Communications Letters*, 6(3):398–401, June 2017.

[AZZ$^+$17] M. AbdelBaky, M. Zou, A. R. Zamani, E. Renart, J. Diaz-Montes, and M. Parashar. Computing in the Continuum: Combining Pervasive Devices and Services to Support Data-Driven Applications. In *IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 1815–1824, June 2017.

[B$^+$15] Massimo Bertoncini et al. Next Generation Data Centers Business Models Enabling Multi-Resource Integration for Smart City Optimized Energy Efficiency. In *ACM e-Energy*, pages 247–252, 2015.

[BAB12] Anton Beloglazov, Jemal Abawajy, and Rajkumar Buyya. Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing. *Future Generation Computer Systems*, 28(5):755 – 768, 2012.

[BCAC$^+$13] Daniel Balouek, Alexandra Carpen-Amarie, Ghislain Charrier, Frédéric Desprez, Emmanuel Jeannot, Emmanuel Jeanvoine, Adrien Lèbre, David Margery, Nicolas Niclausse, Lucas Nussbaum, Olivier Richard, Christian Pérez, Flavien Quesnel, Cyril Rohr, and Luc Sarzyniec. Adding virtualization capabilities to the Grid'5000 testbed. In IvanI. Ivanov, Marten Sinderen, Frank Leymann, and Tony Shan, editors, *Cloud Computing and Services Science*, volume 367 of *Communications in Computer and Information Science*. Springer International Publishing, 2013.

[BCH13] Luiz André Barroso, Jimmy Clidaras, and Urs Hölzle. The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines (Second Edition). *Synthesis Lectures on Computer Architecture*, 8(3):1–156, 2013.

[BCN06] M. Bennett, K. Christensen, and B. Nordman. Improving The Energy Efficiency Of Ethernet: Adaptive Link Rate Proposal. Ethernet Alliance White Paper, 2006.

[BJKT16] S. Bosse, N. Jamous, F. Kramer, and K. Turowski. Introducing Greenhouse Emissions in Cost Optimization of Fault-Tolerant Data Center Design. In *IEEE Conference on Business Informatics (CBI)*, volume 01, pages 163–172, Aug 2016.

[BOÅ+12] Torsten Blochwitz, Martin Otter, Johan Åkesson, et al. Functional mockup interface 2.0: The standard for tool independent exchange of simulation models. In *International Modelica Conference*, pages 173–184, Munich, Germany, 2012.

[Cal] California Institute of Technology. Montage. http://montage.ipac.caltech.edu/.

[CHCC13] H. Chen, C. Hankendi, M. C. Caramanis, and A. K. Coskun. Dynamic server power capping for enabling data center participation in power markets. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 122–129, 2013.

[Cis13] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2012–2017. White paper, May 2013.

[Cis14] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2013–2018. White paper, June 2014.

[Cis15] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2014–2019. White paper, May 2015.

[Cis16] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2015–2020. White paper, June 2016.

[Cis17] Cisco. Cisco Visual Networking Index: Forecast and Methodology, 2016–2021. White paper, June 2017.

[Cis19] Cisco. Cisco Visual Networking Index: Forecast and Trends, 2017–2022. White paper, February 2019.

[CJLF16] X. Chen, L. Jiao, W. Li, and X. Fu. Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing. *IEEE/ACM Transactions on Networking*, 24(5):2795–2808, October 2016.

[CK99] Chandra Chekuri and Sanjeev Khanna. On Multi-Dimensional Packing Problems. In *Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 185–194, 1999.

[Clo19] CloudStack. Administration Guide. `http://docs.cloudstack.apache.org/projects/cloudstack-administration/en/latest/virtual_machines.html`, accessed September 2019.

[CM16] A. Chatzipapas and V. Mancuso. Measurement-based coalescing control for 802.3az. In *IFIP Networking Conference (Networking) and Workshops*, pages 270–278, 2016.

[CMG09] M. Cha, A. Mislove, and K. P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *International conference on World wide web*, 2009.

[CMN09] Luca Chiaraviglio, Marco Mellia, and Fabio Neri. Energy-aware backbone networks: a case study. In *IEEE International Conference on Communications (ICC) Workshops*, pages 1–5, 2009.

[CRN+10] K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J.A. Maestro. IEEE 802.3az: the road to energy efficient ethernet. *IEEE Communications Magazine*, 48(11):50–56, 2010.

[DIg] DIgSILENT PowerFactory. Official website. `https://www.digsilent.de/en/powerfactory.htm`.

[DK13] Christina Delimitrou and Christos Kozyrakis. Paragon: QoS-aware Scheduling for Heterogeneous Datacenters. In *ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 77–88, New York, NY, USA, 2013.

[DMN+17] Thomas Dandres, Reza Farrahi Moghaddam, Kim Khoa Nguyen, Yves Lemieux, Réjean Samson, and Mohamed Cheriet. Consideration of marginal electricity in real-time minimization of distributed data centre emissions. *Journal of Cleaner Production*, 143:116 – 124, 2017.

[Ene18] Enerdata. Global Energy Statistical Yearbook 2018. `https://yearbook.enerdata.net/`, 2018.

[Eps16] Alex Epstein. The Truth About Apple's '100% Renewable' Energy Usage. Forbes, January 2016.

[FFKR15] A. Francini, S. Fortune, T. Klein, and M. Ricca. A low-cost methodology for profiling the power consumption of network equipment. *IEEE Communications Magazine*, 53(5):250–256, May 2015.

[FMI] FMI++ PowerFactor FMU export Utility. Official website. `https://sourceforge.net/projects/powerfactory-fmu/`.

[FPY+09] X. Feng, W. Peterson, F. Yang, G. Wickramasekara, and J. Finney. Smarter grids are more efficient. ABB review, 2009.

[FTK14] Dror G. Feitelson, Dan Tsafrir, and David Krakov. Experience with using the Parallel Workloads Archive. *Journal of Parallel and Distributed Computing*, 74(10):2967 – 2982, 2014.

[G+19] Carla Gomes et al. Computational Sustainability: Computing for a Better World and a Sustainable Future. *Commununications of the ACM*, 62(9):56–65, 2019.

[GCN05] C. Gunaratne, K. Christensen, and B. Nordman. Managing energy consumption costs in desktop PCs and LAN switches with proxying, split TCP connections, and scaling of link speed. *Int. Journal of Network Management*, 15(5):297–310, 2005.

[GFKR15] Yashar Ghiassi-Farrokhfal, Srinivasan Keshav, and Catherine Rosenberg. Toward a realistic performance analysis of storage systems in smart grids. *IEEE Transactions on Smart Grid*, 6(1):402–410, 2015.

[GHT17] J. Guo, G. Hug, and O. Tonguz. Impact of Communication Delay on Asynchronous Distributed Optimal Power Flow Using ADMM. In *IEEE SmartGridComm*, 2017.

[GKL+13] Íñigo Goiri, William Katsak, Kien Le, Thu D Nguyen, and Ricardo Bianchini. Parasol and greenswitch: managing datacenters powered by renewable energy. In *ACM SIGARCH Computer Architecture News*, volume 41, pages 51–64. ACM, 2013.

[Goo] Google. Google Drive. "`https://www.google.com/drive/`".

[GP16] H. Goudarzi and M. Pedram. Hierarchical SLA-Driven Resource Management for Peak Power-Aware and Energy-Efficient Operation of a Cloud Datacenter. *IEEE Transactions on Cloud Computing*, 4(2):222–236, 2016.

[Gre11] How dirty is your data? Greenpeace report, 2011.

[Gre17] Clicking Green: who is winning the race to build a green Internet. Greenpeace report, 2017.

[GRS12] Dhiman Gaurav, Ayoub Raid, and Rosing Tajana S. *Energy and Thermal Aware Scheduling in Data Centers*, chapter 11, pages 301–337. Wiley, 2012.

[GWMR17] M. Ghamkhari, A. Wierman, and H. Mohsenian-Rad. Energy Portfolio Optimization of Data Centers. *IEEE Transactions on Smart Grid*, 8(4):1898–1910, 2017.

[Had] The Apache Hadoop Project. `http://www.hadoop.org`.

[HDMT18] Chang-Hong Hsu, Qingyuan Deng, Jason Mars, and Lingjia Tang. SmoothOperator: Reducing Power Fragmentation and Improving Power Utilization in Large-scale Datacenters. In *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 535–548, 2018.

[HHD+10] Shengsheng Huang, Jie Huang, Jinquan Dai, Tao Xie, and Bo Huang. The HiBench benchmark suite: Characterization of the MapReduce-based data analysis. In *IEEE Data Engineering Workshops (ICDEW)*, pages 41–51, 2010.

[HLL+14] Ward Van Heddeghem, Sofie Lambert, Bart Lannoo, Didier Colle, Mario Pickavet, and Piet Demeester. Trends in worldwide ICT electricity consumption from 2007 to 2012. *Computer Communications*, 50:64 – 76, 2014.

[IEE19] IEEE PES AMPS DSAS Test Feeder Working Group. European Low Voltage Test Feeder. `http://sites.ieee.org/pes-testfeeders/resources/`, Accessed September 2019.

[IET98] IETF. OSPF Version 2. RFC 2328`https://tools.ietf.org/html/rfc2328`, 1998.

[Int19] International Energy Agency. Global energy & co2 status report – the latest trends in energy and emissions in 2018. https://webstore.iea.org/global-energy-co2-status-report-2018, 2019.

[ISO16] Information technology – Data centres – Key performance indicators – Part 2: Power usage effectiveness (PUE). ISO/IEC 30134-2:2016, 2016.

[IT12] Yury Izrailevsky and Ariel Tseitlin. The Netflix Simian Army. Netflix Tech Blog, `https://netflixtechblog.com/the-netflix-simian-army-16e57fbab116`, 2012.

[JAV+14] Fatemeh Jalali, Robert Ayre, Arun Vishwanath, Kerry Hinton, Tansu Alpcan, and Rodney S. Tucker. Energy Consumption of Content Distribution from Nano Data Centers versus Centralized Data Centers. *SIGMETRICS Performance Evaluation Review*, 42(3):49–54, 2014.

[Jef12] Brian Jeff. Big.LITTLE system architecture from ARM: saving power through heterogeneous multiprocessing and task context migration. In *Annual Design Automation Conference (DAC)*, pages 1143–1146, 2012.

[JHA+16] Fatemeh Jalali, Kerry Hinton, Robert Ayre, Tansu Alpcan, and Rodney S Tucker. Fog Computing May Help to Save Energy in Cloud Computing. *IEEE Journal on Selected Areas in Communications*, 34(5):1728–1739, 2016.

[JSW+16] A. Jin, W. Song, P. Wang, D. Niyato, and P. Ju. Auction Mechanisms Toward Efficient Resource Sharing for Cloudlets in Mobile Cloud Computing. *IEEE Transactions on Services Computing*, 9(6):895–909, Nov 2016.

[KL16] Y. W. Kuo and C. L. Li. Design of long range low power sensor node for the last mile of IoT. In *IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, pages 1–2, 2016.

[KTB17] Atefeh Khosravi, Adel Nadjaran Toosi, and Rajkumar Buyya. Online virtual machine migration for renewable energy usage maximization in geographically distributed cloud data centers. *Concurrency and Computation: Practice and Experience*, 29(18), 2017.

[LEGE14] A.Q. Lawey, T.E.H. El-Gorashi, and J.M.H. Elmirghani. Distributed Energy Efficient Clouds Over Core Networks. *Journal of Lightwave Technology*, 32(7):1261–1281, April 2014.

[Len16] Lionel Lenôtre. A Strategy for Parallel Implementations of Stochastic Lagrangian Simulation. In *Monte Carlo and Quasi-Monte Carlo Methods*, pages 507–520. Springer, 2016.

[LS17] Yuhua Lin and Haiying Shen. EAFR: An Energy-Efficient Adaptive File Replication System in Data-Intensive Clusters. *IEEE Transactions on Parallel and Distributed Systems*, 28(4), 2017.

[MBM+18] C. X. Mavromoustakis, J. M. Batalla, G. Mastorakis, E. Markakis, and E. Pallis. Socially oriented edge computing for energy awareness in iot architectures. *IEEE Communications Magazine*, 56(7):139–145, July 2018.

[MDHS09] Todd Mytkowicz, Amer Diwan, Matthias Hauswirth, and Peter F. Sweeney. Producing Wrong Data Without Doing Anything Obviously Wrong! In *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. ACM, 2009.

[Mil13] M. Mills. The Cloud Begins With Coal - Big Data, Big Networks, Big Infrastructure, and Big Power. Technical report, Digital Power Group, Aug. 2013.

[MKB18] Redowan Mahmud, Ramamohanarao Kotagiri, and Rajkumar Buyya. *Fog Computing: A Taxonomy, Survey and Future Directions*, pages 103–130. Springer, 2018.

[MNY+18] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos. A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges. *IEEE Communications Surveys Tutorials*, 20(1):416–464, Firstquarter 2018.

[NCB] NCBI National Center for Biotechnology Information. Basic Local Alignment Search Tool (Blast). `https://blast.ncbi.nlm.nih.gov/Blast.cgi`.

[NCR+18] Marco AS Netto, Rodrigo N Calheiros, Eduardo R Rodrigues, Renato LF Cunha, and Rajkumar Buyya. HPC Cloud for Scientific and Business Applications: Taxonomy, Vision, and Research Challenges. *ACM Computing Surveys*, 51(1):8, 2018.

[NHQ+15] Nam Pham Ngoc, Thanh Nguyen Huu, Trong Vu Quang, Vu Tran Hoang, Huong Truong Thu, Phuoc Tran-Gia, and Christian Schwartz. A new power profiling method and power scaling mechanism for energy-aware NetFPGA gigabit router. *Computer Networks*, 78:4 – 25, 2015. Special Issue: Green Communications.

[Nor12] William D. Nordhaus. *Carbon Taxes to Move Toward Fiscal Sustainability*, pages 208–214. Columbia University Press, 2012.

[NPI+08] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall. Reducing Network Energy Consumption via Sleeping and Rate-Adaptation. In *USENIX Symp. On Netw. Systems Design & Implementation (NSDI)*, pages 323–336, 2008.

[ns3] ns3 network simulator. `http://www.nsnam.org`.

[OLGLLP11] Anne-Cécile Orgerie, Laurent Lefèvre, Isabelle Guérin-Lassous, and Dino Lopez Pacheco. Ecofen: an end-to-end energy cost model and simulator for evaluating power consumption in large-scale networks. In *SustaInet: First International Workshop on Sustainable Internet and Internet for Sustainability (in conjunction with WoWMoM)*, 2011.

[OM] Openmodelica. `https://openmodelica.org`.

[Ope] OpenStack Project. OpenStack: The open source cloud operating system. `openstack.org`.

[Ope19] OpenStack Configuration Reference. Nova scheduler. `https://docs.openstack.org/mitaka/config-reference/compute/scheduler.html`, accessed September 2019.

[Pho] Photovolta project - university of nantes, france. `http://photovolta2.univ-nantes.fr`.

[PP18] Olusogo Popoola and Bernardi Pranggono. On energy consumption of switch-centric data center networks. *The Journal of Supercomputing*, 74(1):334–369, Jan 2018.

[R+17] I. Ramezanipour et al. Decision Error Probability in a Two-stage Communication Network for Smart Grids with Imperfect Links. In *IEEE SmartGridComm*, 2017.

[Res16] Lux Research. Coal Computing: How Companies Misunderstand Their Dirty Data Centers. White paper, 2016.

[Ric] Rice University Bidding System. RUBiS.

[Rig15] Right Scale. State of the Cloud Report. `https://assets.rightscale.com/uploads/pdfs/RightScale-2015-State-of-the-Cloud-Report.pdf`, 2015.

[RLK14] Ashikur Rahman, Xue Liu, and Fanxin Kong. A Survey on Geographic Load Balancing Based Data Center Power Management in the Smart Grid Environment. *IEEE Communications Surveys & Tutorials*, 16(1):214–233, 2014.

[RS16] L. Rosyidi and R. F. Sari. Energy harvesting aware protocol for 802.11-based Internet of Things network. In *IEEE Region 10 Conference (TENCON)*, pages 1325–1328, 2016.

[RWH11] Charles Reiss, John Wilkes, and Joseph L. Hellerstein. Google cluster-usage traces: format + schema. Technical report, Google Inc., Mountain View, CA, USA, November 2011.

[San18] Sandvine. The Global Internet Phenomena Report. `https://www.sandvine.com/phenomena`, October 2018.

[Sea12] Seagate. Desktop HDD specification sheet. http://www.seagate.com/staticfiles/docs/pdf/datasheet/disc/desktop-hdd-data-sheet-ds1770-1-1212us.pdf, 2012.

[SGSB+15] Pablo Serrano, Andres Garcia-Saavedra, Giuseppe Bianchi, Albert Banchs, and Arturo Azcorra. Per-frame Energy Consumption in 802.11 Devices and Its Implication on Modeling and Design. *IEEE/ACM Transaction on Netwing*, 23(4):1243–1256, 2015.

[Sima] Simgrid. https://simgrid.org.

[Simb] SimGrid-FMI. https://framagit.org/simgrid/simgrid-FMI.

[SNSC18] G. Suciu, L. Necula, V. Suciu, and Y. Curtmola. Cloud-based platform for enhancing energy consumption awareness and substantiating the adoption of energy efficiency measures within smes. In *2018 14th International Wireless Communications Mobile Computing Conference (IWCMC)*, pages 1018–1023, June 2018.

[ST08] G Jeffrey Snyder and Eric S Toberer. Complex thermoelectric materials. *Nature materials*, 7(2):105–114, 2008.

[Sta18] Statista. Worldwide internet user penetration from 2014 to 2021, 2018.

[STM14] Parnia Samimi, Youness Teimouri, and Muriati Mukhtar. A combinatorial double auction resource allocation model in cloud computing. *Information Sciences*, 2014.

[SVZR11] V. Sivaraman, A. Vishwanath, Zhi Zhao, and C. Russell. Profiling per-packet and per-byte energy consumption in the NetFPGA Gigabit router. In *2011 Conference on Computer Communications Workshops (INFOCOM Workshops)*, pages 331–336, 2011.

[Tay12] Michael B. Taylor. Is Dark Silicon Useful? Harnessing the Four Horsemen of the Coming Dark Silicon Apocalypse. In *Annual Design Automation Conference (DAC)*, page 1131–1136, 2012.

[The18] The Shift Project. Lean ICT, Pour une sobriété numérique. https://theshiftproject.org/article/pour-une-sobriete-numerique-rapport-shift/, October 2018.

[TKBL12] M. Tighe, G. Keller, M. Bauer, and H. Lutfiyya. DCSim: A data centre simulation tool for evaluating dynamic virtualized resource management. In *Workshop on Systems Virtualization Management (colocated with CNSM)*, pages 385–392, Oct 2012.

[Top19] TOP500. https://www.top500.org, 2019.

[TPC00] TPC Benchmark W (Web Commerce) Specification. version 1.0.1, 2000. www.tpc.org/tpcw.

[Uni18] International Telecommunication Union. Measuring the Information Society Report. https://www.itu.int/en/mediacentre/Pages/2018-PR40.aspx, 2018.

[VP16] M. Vautier and O. Philippot. Is "software eco-design" a solution to reduce the environmental impact of electronic equipments? In *Electronics Goes Green 2016+ (EGG)*, pages 1–6, 2016.

[VPC+15] Julien Vaubourg, Yannick Presse, Benjamin Camus, Christine Bourjot, Laurent Ciarletta, Vincent Chevrier, Jean-Philippe Tavella, and Hugo Morais. Multi-agent multi-model simulation of smart grids in the MS4SG project. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, 2015.

[VWB+16] Blesson Varghese, Nan Wang, Sakil Barbhuiya, Peter Kilpatrick, and Dimitrios S. Nikolopoulos. Challenges and Opportunities in Edge Computing. *CoRR*, abs/1609.01967, 2016.

[WB14] R. Wolski and J. Brevik. Using Parametric Models to Represent Private Cloud Workloads. *IEEE Transactions on Services Computing*, 7(4):714–725, 2014.

[WCP+15] Usman Wajid, Cinzia Cappiello, Pierluigi Plebani, Barbara Pernici, Nikolay Mehandjiev, Monica Vitali, Michael Gienger, Konstantinos Kavoussanakis, David Margery, David García-Pérez, and Pedro Sampaio. On Achieving Energy Efficiency and Reducing CO2 Footprint in Cloud Computing. *IEEE Transactions on Cloud Computing*, PP(99):14, July 2015.

[Wir] Wireshark. `https://www.wireshark.org/`.

[WNP11] He Wu, Sidharth Nabar, and Radha Poovendran. An Energy Framework for the Network Simulator 3 (NS-3). In *International Conference on Simulation Tools and Techniques (SIMUTools)*, pages 222–230, 2011.

[Wor19] Worldometers. World Population: Past, Present, and Future. `https://www.worldometers.info/world-population/`, 2019.

[WTM14] Hui Wang, Huaglory Tianfield, and Quentin Mair. Auction Based Resource Allocation in Cloud Computing. *Multiagent Grid Systems*, 10(1):51–66, January 2014.

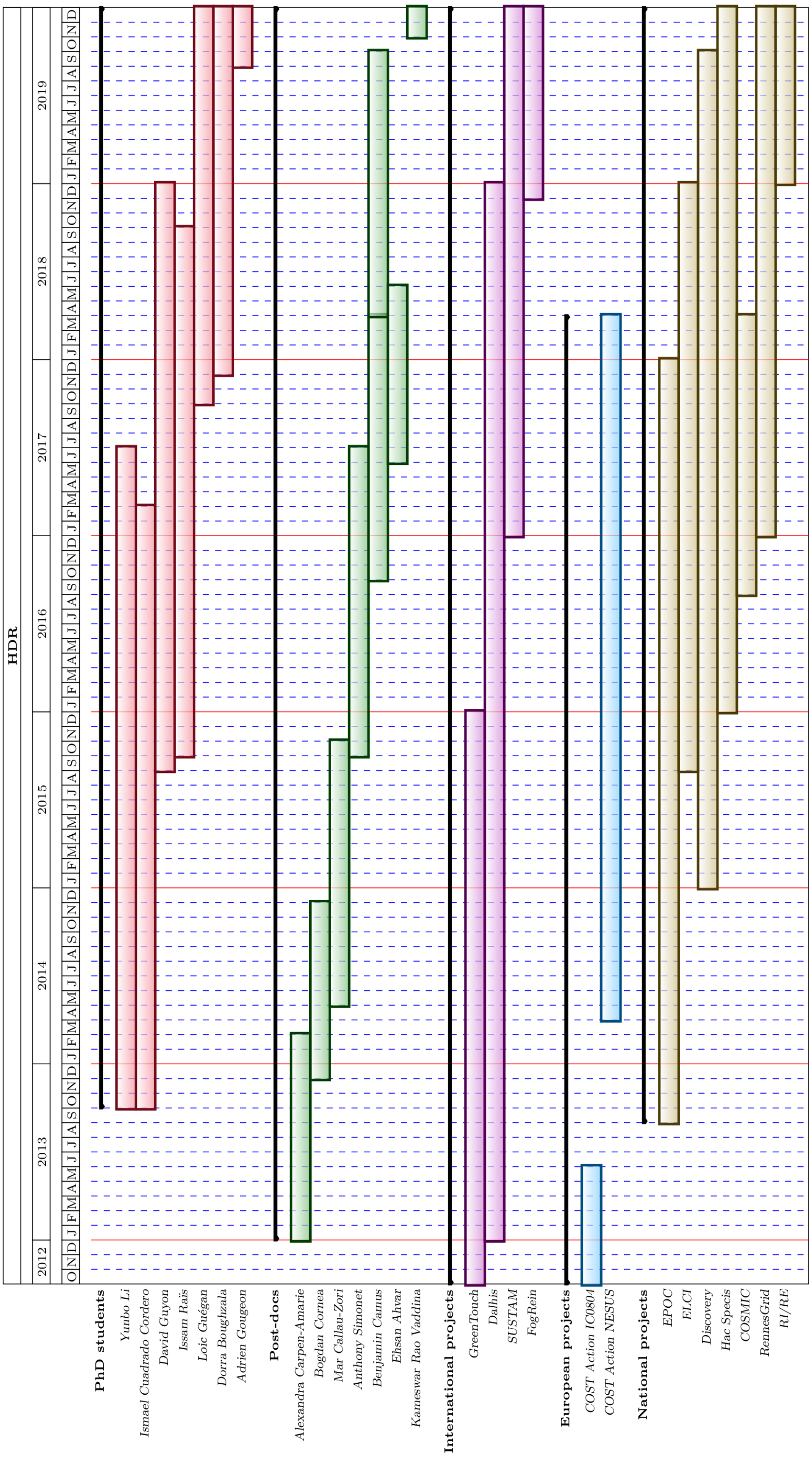
[XD16] H. Xu, J. Sun and C. Dent. The Coordinated Voltage Control Meets Imperfect Communication System. In *IEEE PES ISGT-Europe*, 2016.

[YLH+18] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang. A Survey on the Edge Computing for the Internet of Things. *IEEE Access*, 6:6900–6919, 2018.

[Yue91] Minyi Yue. A simple proof of the inequality $FFD(L) \leq \frac{11}{9}OPT(L) + 1, \forall L$ for the FFD bin-packing algorithm. *Acta mathematicae applicatae sinica*, 7(4):321–331, 1991.

[YWL+14] F. Yang, S. Wang, J. Li, Z. Liu, and Q. Sun. An overview of internet of vehicles. *China Communications*, 11(10):1–15, 2014.

Gantt from October 2012 to January 2020



As for the future,
your task is not
to foresee it,
but to enable it.
Antoine de Saint-Exupéry

List of coauthors from October 2012 to January 2020

Deb Agarwal	Emmanuel Agullo	Ehsan Ahvar
Francisco Almeida	Pedro Alonso	Betsegaw Lemma Amersho
Gabriel Antoniu	Michel Bagein	Daniel Balouek-Thomert
Jorge Barbosa	Nicolas Beldiceanu	Anne Benoit
Françoise Berthoud	Marin Bertier	Jean-Baptiste Blanc-Rouchossé
Vicente Blanco	Anne Blavette	Mathilde Boutigny
Ivona Brandic	Mar Callau-Zori	Benjamin Camus
Alexandra Carpen-Amarie	Henri Casanova	Michele Chincoli
Bogdan Cornea	Tom Cornebize	Samuel Cremer
Felix Cuadrado	Ismael Cuadrado-Cordero	Georges Da Costa
Vincenzo De Maio	Hermann De Meer	Ewa Deelman
Augustin Degomme	Frédéric Desprez	Marcos Dias de Assunção
Djawida Dib	Lars Dittmann	Manuel Dolz
Matthieu Dorier	Eric Drezet	Fanny Dufossé
Barbara Dumas Feris	Anne Elster	Gilles Fedak
Pascal Felber	Rafael Ferreira da Silva	Andreas Fischer
Gareth Francis	Neki Frasheri	Sebastien Fremal
Jaime Galan-Jimenez	Alberto Garcia	Victor Garcia
Alfonso Gazo Cervero	Luc Giraud	Philippe Gravey
Loic Guegan	Amina Guermouche	Joel Guerrero
David Guyon	Sabbir Hasan	Timothée Haudebourg
Christian Heinrich	Sasha Hunold	Shadi Ibrahim
Claude Jard	Mateusz Jarus	Helen Karatza
Kostas Kavoussanakis	Mascha Kurpicz	Sébastien Lafond
Stéphane Lanteri	Alexey Lastovetsky	Thomas Ledoux
Laurent Lefèvre	Adrien Lebre	Arnaud Legrand
Yunbo Li	Didier Lime	Dino Lopez Pacheco
Gilles Madi-Wamba	Ravi Reddy Manumachu	Gilles Marait
David Margery	Toni Mastelic	Ilias Mavridis
Jean-Marc Menaud	Thierry Monteil	Paolo Monti
Pascal Morel	Christine Morin	Michel Morvan
Marie-Laure Moulinard	Benson Muite	Ariel Oleksiak
Charaka Palansuriya	Manish Parashar	Jonathan Pastor
Louis-Francois Pau	Jean-Louis Pazat	Wojciech Piatek
Guillaume Pierre	Jean-Marc Pierson	Chris Phillips
Louis Poirel	Radu Prodan	Flavien Quesnel
Martin Quinson	Issam Raïs	Sergio Ricciardi
Myriana Rifai	Ivan Rodero	Olivier Roux
Jonathan Rouzaud-Cornabas	Lavinia Samoila	Ammar Sharaiha
Rémi Sharrock	Enida Sheme	Anthony Simonet
Anita Sobe	Georgios Stavrinides	Patricia Stolf
Frédéric Suter	Ryan Tanaka	Cédric Tedeschi
Tuan Trinh	Luca Valcarenghi	Sebastien Varrette
Orçun Yildiz	*To be continued...*	

Publications from October 2012 to January 2020

[1] Agullo, E., Giraud, L., Lanteri, S., Marait, G., Orgerie, A.-C., and Poirel, L. Energy Analysis of a Solver Stack for Frequency-Domain Electromagnetics. In *PDP: Euromicro International Conference on Parallel, Distributed and Network-Based Processing* (Pavia, Italy, Feb. 2019), pp. 385–391.

[2] Ahvar, E., Orgerie, A.-C., and Lebre, A. Estimating Energy Consumption of Cloud, Fog and Edge Computing Infrastructures. *IEEE Transactions on Sustainable Computing* (Apr. 2019), 1–12.

[3] Almeida, F., Dias de Assuncao, M., Barbosa, J., Blanco, V., Brandic, I., Da Costa, G., Dolz, M. F., Elster, A. C., Jarus, M., Karatza, H., Lefèvre, L., Mavridis, I., Oleksiak, A., Orgerie, A.-C., and Pierson, J.-M. Energy Monitoring as an Essential Building Block Towards Sustainable Ultrascale Systems. *Sustainable Computing: Informatics and Systems 17* (Mar. 2018), 27–42.

[4] Bagein, M., Barbosa, J., Blanco, V., Brandic, I., Cremer, S., Fremal, S., Karatza, H., Lefèvre, L., Mastelic, T., Oleksiak, A., Orgerie, A.-C., Stavrinides, G. L., and Varrette, S. Energy Efficiency for Ultrascale Systems: Challenges and Trends from Nesus Project. *Supercomputing frontiers and innovations 2*, 2 (Sept. 2015), 105–131.

[5] Beldiceanu, N., Dumas Feris, B., Gravey, P., Hasan, M. S., Jard, C., Ledoux, T., Li, Y., Lime, D., Madi-Wamba, G., Menaud, J.-M., Morel, P., Morvan, M., Moulinard, M.-L., Orgerie, A.-C., Pazat, J.-L., Roux, O., and Sharaiha, A. The EPOC project: Energy Proportional and Opportunistic Computing system. In *International Conference on Smart Cities and Green ICT Systems (SmartGreens)* (Lisbonne, Portugal, May 2015). 15397.

[6] Beldiceanu, N., Dumas Feris, B., Gravey, P., Hasan, M. S., Jard, C., Ledoux, T., Li, Y., Lime, D., Madi-Wamba, G., Menaud, J.-M., Morel, P., Morvan, M., Moulinard, M.-L., Orgerie, A.-C., Pazat, J.-L., Roux, O. H., and Sharaiha, A. Towards energy-proportional Clouds partially powered by renewable energy. *Computing 99*, 1 (Jan. 2017), 3–22.

[7] Benoit, A., Lefèvre, L., Orgerie, A.-C., and Raïs, I. Shutdown Policies with Power Capping for Large Scale Computing Systems. In *Euro-Par: International European Conference on Parallel and Distributed Computing* (Santiago de Compostela, Spain, Aug. 2017), F. F. Rivera, T. F. Pena, and J. C. Cabaleiro, Eds., vol. 10417, pp. 134 – 146.

[8] Benoit, A., Lefèvre, L., Orgerie, A.-C., and Raïs, I. Reducing the energy consumption of large scale computing systems through combined shutdown policies with multiple constraints. *International Journal of High Performance Computing Applications 32*, 1 (Jan. 2018), 176–188.

[9] Berthoud, F., Drezet, E., Lefèvre, L., and Orgerie, A.-C. La déferlante des données. *Interstices* (July 2015).

[10] Berthoud, F., Drezet, E., Lefèvre, L., and Orgerie, A.-C. Le syndrome de l'obésiciel: des applications énergivores. *Interstices* (July 2015).

[11] Berthoud, F., Drezet, E., Lefèvre, L., and Orgerie, A.-C. L'épidémie du smartphone : prolifération et dissémination des composants électroniques. *Interstices* (June 2015).

[12] Berthoud, F., Drezet, E., Lefèvre, L., and Orgerie, A.-C. Sciences du numérique et développement durable : des liens complexes. *Interstices* (June 2015).

[13] Callau-Zori, M., Samoila, L., Orgerie, A.-C., and Pierre, G. An experiment-driven energy consumption model for virtual machine management systems. *Sustainable Computing: Informatics and Systems 18* (June 2018), 163–174.

[14] Camus, B., Blavette, A., Dufossé, F., and Orgerie, A.-C. Harnessing the geographical flexibility of distributed computing clouds for cooperative self-consumption. In *ISGT-Europe: IEEE PES Innovative Smart Grid Technologies Conference Europe* (Sarajevo, Bosnia and Herzegovina, Oct. 2018), pp. 1–6.

[15] Camus, B., Blavette, A., Dufossé, F., and Orgerie, A.-C. Self-Consumption Optimization of Renewable Energy Production in Distributed Clouds. In *IEEE International Conference on Cluster Computing (Cluster)* (Belfast, United Kingdom, Sept. 2018), pp. 1–11.

[16] Camus, B., Blavette, A., Orgerie, A.-C., and Blanc-Rouchossé, J.-B. Co-simulation of an electrical distribution network and its supervision communication network. In *CCNC: IEEE Consumer Communications & Networking Conference* (Las Vegas, United States, Jan. 2020).

[17] Camus, B., Dufossé, F., Blavette, A., Quinson, M., and Orgerie, A.-C. Network-aware energy-efficient virtual machine management in distributed Cloud infrastructures with on-site photovoltaic production. In *SBAC-PAD: International Symposium on Computer Architecture and High Performance Computing* (Lyon, France, Sept. 2018), pp. 1–8.

[18] Camus, B., Dufossé, F., and Orgerie, A.-C. A stochastic approach for optimizing green energy consumption in distributed clouds. In *International Conference on Smart Cities and Green ICT Systems (SmartGreens)* (Porto, Portugal, Apr. 2017).

[19] Camus, B., Dufossé, F., and Orgerie, A.-C. The SAGITTA approach for optimizing solar energy consumption in distributed clouds with stochastic modeling. In *Smart Cities, Green Technologies, and Intelligent Transport Systems*. Dec. 2018, pp. 1–25.

[20] Camus, B., Orgerie, A.-C., and Quinson, M. Co-simulation of FMUs and Distributed Applications with SimGrid. In *SIGSIM-PADS: SIGSIM Principles of Advanced Discrete Simulation* (Rome, Italy, May 2018), ACM, pp. 145–156.

[21] Carpen-Amarie, A., Dib, D., Orgerie, A.-C., and Pierre, G. Towards Energy-Aware IaaS-PaaS Co-design. In *SmartGreens: International Conference on Smart Grids and Green IT Systems* (Barcelone, Spain, Apr. 2014).

[22] Carpen-Amarie, A., Orgerie, A.-C., and Morin, C. Experimental Study on the Energy Consumption in IaaS Cloud Environments. In *IEEE/ACM International Conference on Utility and Cloud Computing (UCC)* (Dresden, Germany, Dec. 2013).

[23] Cornea, B. F., Orgerie, A.-C., and Lefèvre, L. Studying the energy consumption of data transfers in Clouds: the Ecofen approach. In *CloudNet: IEEE International Conference on Cloud Networking* (Luxembourg, Luxembourg, Oct. 2014).

[24] Cuadrado-Cordero, I., Cuadrado, F., Phillips, C., Orgerie, A.-C., and Morin, C. Microcities: a Platform based on Microclouds for Neighborhood Services. Research Report RR-8885, inria, Feb. 2016.

[25] Cuadrado Cordero, I., Cuadrado, F., Phillips, C., Orgerie, A.-C., and Morin, C. Microcities: a Platform based on Microclouds for Neighborhood Services. In *International Conference on Algorithms and Architectures for Parallel (ICA3PP)* (Granada, Spain, Dec. 2016), vol. 10048, Springer, pp. 192–202.

[26] Cuadrado-Cordero, I., Orgerie, A.-C., and Menaud, J.-M. Comparative Experimental Analysis of the Quality-of-Service and Energy-Efficiency of VMs and Containers' Consolidation for Cloud Applications. In *International Conference on Software, Telecommunications and Computer Networks (SoftCOM)* (Split, Croatia, Sept. 2017), pp. 1–6.

[27] Cuadrado Cordero, I., Orgerie, A.-C., and Morin, C. GRaNADA: A Network-Aware and Energy-Efficient PaaS Cloud Architecture. In *IEEE International Conference on Green Computing and Communications (GreenCom)* (Sydney, Australia, Dec. 2015).

[28] Cuadrado-Cordero, I., Orgerie, A.-C., and Morin, C. Incentives for Mobile Cloud Environments through P2P Auctions. In *CloudNet: IEEE International Conference on Cloud Networking* (Pisa, Italy, Oct. 2016).

[29] Desprez, F., Ibrahim, S., Lebre, A., Orgerie, A.-C., Pastor, J., and Simonet, A. Energy-Aware Massively Distributed Cloud Facilities: The DISCOVERY Initiative. In *IEEE International Conference on Green Computing and Communications (GreenCom)* (Sydney, Australia, Dec. 2015), IEEE International Conference on Green Computing and Communications (GreenCom), pp. 476 – 477.

[30] Dorier, M., Yildiz, O., Ibrahim, S., Orgerie, A.-C., and Antoniu, G. On the energy footprint of I/O management in Exascale HPC systems. *Future Generation Computer Systems 62* (Mar. 2016), 17–28.

[31] Ferreira Da Silva, R., Orgerie, A.-C., Casanova, H., Tanaka, R., Deelman, E., and Suter, F. Accurately Simulating Energy Consumption of I/O-intensive Scientific Workflows. In *International Conference on Computational Science (ICCS)* (Faro, Portugal, June 2019), pp. 138–152.

[32] Gazo Cervero, A., Chincoli, M., Dittmann, L., Fischer, A., Garcia, A. E., Galán-Jiménez, J., Lefèvre, L., de Meer, H., Monteil, T., Monti, P., Orgerie, A.-C., Pau, L.-F., Phillips, C., Ricciardi, S., Sharrock, R., Stolf, P., Trinh, T., and Valcarenghi, L. Green Wired Networks. In *Large-Scale Distributed Systems and Energy Efficiency*. Wiley, Mar. 2015, pp. 41–80.

[33] Guegan, L., Amersho, B. L., Orgerie, A.-C., and Quinson, M. A Large-Scale Wired Network Energy Model for Flow-Level Simulations. In *International Conference on Advanced Information Networking and Applications (AINA)* (Matsue, Japan, Mar. 2019), vol. 926, Springer, pp. 1047–1058.

[34] Guegan, L., and Orgerie, A.-C. Estimating the end-to-end energy consumption of low-bandwidth IoT applications for WiFi devices. In *CloudCom: IEEE International Conference on Cloud Computing Technology and Science* (Sydney, Australia, Dec. 2019).

[35] Guermouche, A., and Orgerie, A.-C. Experimental analysis of vectorized instructions impact on energy and power consumption under thermal design power constraints. Tech. rep., June 2019. working paper or preprint.

[36] Guyon, D., Orgerie, A.-C., and Morin, C. Energy-efficient User-oriented Cloud Elasticity for Data-driven Applications. In *IEEE International Conference on Green Computing and Communications (GreenCom)* (Sydney, Australia, Dec. 2015).

[37] Guyon, D., Orgerie, A.-C., and Morin, C. GLENDA: Green Label towards Energy proportioNality for IaaS DAta centers. In *E2DC: International Workshop on Energy Efficient Data Centres (e-Energy Workshop)* (Hong Kong, Hong Kong SAR China, May 2017).

[38] Guyon, D., Orgerie, A.-C., and Morin, C. An Experimental Analysis of PaaS Users Parameters on Applications Energy Consumption. In *IC2E: IEEE International Conference on Cloud Engineering* (Orlando, United States, Apr. 2018), pp. 170–176.

[39] Guyon, D., Orgerie, A.-C., and Morin, C. Energy-Efficient IaaS-PaaS Co-design for Flexible Cloud Deployment of Scientific Applications. In *SBAC-PAD: International Symposium on Computer Architecture and High Performance Computing* (Lyon, France, Sept. 2018), pp. 1–8.

[40] Guyon, D., Orgerie, A.-C., Morin, C., and Agarwal, D. How Much Energy can Green HPC Cloud Users Save? In *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing* (Saint Petersbourg, Russia, Mar. 2017).

[41] Guyon, D., Orgerie, A.-C., Morin, C., and Agarwal, D. Involving Users in Energy Conservation: A Case Study in Scientific Clouds. *International Journal of Grid and Utility Computing* (2018), 1 – 14.

[42] Haudebourg, T., and Orgerie, A.-C. On the Energy Efficiency of Sleeping and Rate Adaptation for Network Devices. In *International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP)* (Helsinki, Finland, Aug. 2017), vol. 10393, pp. 132–146. Best Paper Award.

[43] Heinrich, F. C., Carpen-Amarie, A., Degomme, A., Hunold, S., Legrand, A., Orgerie, A.-C., and Quinson, M. Predicting the Performance and the Power Consumption of MPI Applications With SimGrid. Tech. rep., Jan. 2017. working paper or preprint.

[44] Heinrich, F. C., Cornebize, T., Degomme, A., Legrand, A., Carpen-Amarie, A., Hunold, S., Orgerie, A.-C., and Quinson, M. Predicting the Energy Consumption of MPI Applications at Scale Using a Single Node. In *IEEE Cluster Conference* (Hawaii, United States, Sept. 2017).

[45] Kurpicz, M., Orgerie, A.-C., and Sobe, A. How much does a VM cost? Energy-proportional Accounting in VM-based Environments. In *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing* (Heraklion, Greece, Feb. 2016), p. 8.

[46] Kurpicz, M., Orgerie, A.-C., Sobe, A., and Felber, P. Energy-proportional Profiling and Accounting in Heterogeneous Virtualized Environments. *Sustainable Computing: Informatics and Systems 18* (June 2018), 175–185.

[47] Lebre, A., Pastor, J., Bertier, M., Desprez, F., Rouzaud-Cornabas, J., Tedeschi, C., Orgerie, A.-C., Quesnel, F., and Fedak, G. Beyond The Clouds, How Should Next Generation Utility Computing Infrastructures Be Designed? In *Cloud Computing: Challenges, Limitations and R&D Solutions*, Z. Mahmood, Ed. Springer, Nov. 2014.

[48] Lebre, A., Simonet, A., and Orgerie, A.-C. Deploying Distributed Cloud Infrastructures: Who and at What Cost? In *IEEE International Workshop on Cloud Computing Interclouds, Multiclouds, Federations, and Interoperability (InterCloud)* (Berlin, Germany, Apr. 2016), p. 6.

[49] Li, Y., Orgerie, A.-C., and Menaud, J.-M. Opportunistic Scheduling in Clouds Partially Powered by Green Energy. In *IEEE International Conference on Green Computing and Communications (GreenCom)* (Sydney, Australia, Dec. 2015).

[50] Li, Y., Orgerie, A.-C., and Menaud, J.-M. Balancing the use of batteries and opportunistic scheduling policies for maximizing renewable energy consumption in a Cloud data center. In *PDP: Euromicro International Conference on Parallel, Distributed, and Network-Based Processing* (St Petersburg, Russia, Mar. 2017).

[51] Li, Y., Orgerie, A.-C., Rodero, I., Lemma Amersho, B., Parashar, M., and Menaud, J.-M. End-to-end Energy Models for Edge Cloud-based IoT Platforms: Application to Data Stream Analysis in IoT. *Future Generation Computer Systems 87* (Oct. 2018), 667–678.

[52] Li, Y., Orgerie, A.-C., Rodero, I., Parashar, M., and Menaud, J.-M. Leveraging Renewable Energy in Edge Clouds for Data Stream Analysis in IoT. In *IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)* (Madrid, Spain, May 2017), pp. 186–195.

[53] Madi Wamba, G., Li, Y., Orgerie, A.-C., Beldiceanu, N., and Menaud, J.-M. Cloud workload prediction and generation models. In *SBAC-PAD: International Symposium on Computer Architecture and High Performance Computing* (Campinas, Brazil, Oct. 2017), pp. 89–96.

[54] Madi Wamba, G., Li, Y., Orgerie, A.-C., Beldiceanu, N., and Menaud, J.-M. Green energy aware scheduling problem in virtualized datacenters. In *IEEE International Conference on Parallel and Distributed Systems (ICPADS)* (Shenzen, China, Dec. 2017), pp. 648–655.

[55] Margery, D., Guyon, D., Orgerie, A.-C., Morin, C., Francis, G., Palansuriya, C., and Kavoussanakis, K. A CO2 emissions accounting framework with market-based incentives for Cloud infrastructures. In *SmartGreens: International Conference on Smart Cities and Green ICT Systems* (Porto, Portugal, Apr. 2017).

[56] Oleksiak, A., Lefèvre, L., Alonso, P., Da Costa, G., De Maio, V., Frasheri, N., Garcia, V., Guerrero, J., Lafond, S., Lastovetsky, A., Manumachu, R. R., Muite, B., Orgerie, A.-C., Piatek, W., Pierson, J.-M., Prodan, R., Stolf, P., Sheme, E., and Varrette, S. Energy aware ultrascale systems. In *Ultrascale Computing Systems*. Institution of Engineering and Technology, Jan. 2019, pp. 127–188.

[57] Orgerie, A.-C. Interconnecting Smart Grids and Clouds to save Energy. In *International Conference on Smart Cities and Green ICT Systems (SmartGreens)* (Lisbon, Portugal, May 2015), p. 6.

[58] Orgerie, A.-C. Green Computing and Sustainability. In *Energie et radiosciences - Journées scientifiques URSI France* (Rennes, France, Mar. 2016).

[59] Orgerie, A.-C., Dias de Assuncao, M., and Lefèvre, L. A Survey on Techniques for Improving the Energy Efficiency of Large Scale Distributed Systems. *ACM Computing Surveys 46*, 4 (Dec. 2014), 31.

[60] Orgerie, A.-C., Lemma Amersho, B., Haudebourg, T., Quinson, M., Rifai, M., Lopez Pacheco, D., and Lefèvre, L. Simulation Toolbox for Studying Energy Consumption in Wired Networks. In *CNSM: International Conference on Network and Service Management* (Tokyo, Japan, Nov. 2017), pp. 1–5.

[61] Raïs, I., Balouek-Thomert, D., Orgerie, A.-C., Lefèvre, L., and Parashar, M. Leveraging energy-efficient non-lossy compression for data-intensive applications. In *HPCS: International Conference on High Performance Computing & Simulation* (Dublin, Ireland, July 2019), pp. 1–7.

[62] Raïs, I., Boutigny, M., Lefèvre, L., Orgerie, A.-C., and Benoit, A. Building the Table of Energy and Power Leverages for Energy Efficient Large Scale Systems. In *HPCS: International Conference on High Performance Computing & Simulation* (Orléans, France, July 2018), pp. 284–291.

[63] Raïs, I., Lefèvre, L., Benoit, A., and Orgerie, A.-C. An Analysis of the Feasibility of Energy Harvesting with Thermoelectric Generators on Petascale and Exascale Systems. In *Workshop Optimization of Energy Efficient HPC & Distributed Systems (OPTIM), co-located with HPCS* (Innsbruck, Austria, July 2016).

[64] Raïs, I., Lefèvre, L., Orgerie, A.-C., and Benoit, A. Exploiting the Table of Energy and Power Leverages. In *ICA3PP: International Conference on Algorithms and Architectures for Parallel Processing* (Guangzhou, China, Nov. 2018), pp. 1–10.

[65] Raïs, I., Orgerie, A.-C., and Quinson, M. Impact of Shutdown Techniques for Energy-Efficient Cloud Data Centers. In *ICA3PP: 16th International Conference on Algorithms and Architectures for Parallel Processing* (Granada, Spain, Dec. 2016), pp. 203–210.

[66] Raïs, I., Orgerie, A.-C., Quinson, M., and Lefèvre, L. Quantifying the Impact of Shutdown Techniques for Energy-Efficient Data Centers. *Concurrency and Computation: Practice and Experience 30*, 17 (2018), 1–13.