DATAZERO: DATAcenter With Zero Emission and Robust Management Using Renewable Energy

JEAN-MARC PIERSON[1], (Member, IEEE), **GWILHERM BAUDIC**[1], **STÉPHANE CAUX**[3],
BERK CELIK[2,3], (Member, IEEE), **GEORGES DA COSTA**[1], **LÉO GRANGE**[1], **MARWA HADDAD**[2],
JÉRÔME LECUIVRE[4], **JEAN-MARC NICOD**[2], **LAURENT PHILIPPE**[2], **VERONIKA REHN-SONIGO**[2],
ROBIN ROCHE[2], (Senior Member, IEEE), **GUSTAVO ROSTIROLLA**[1,3], **AMAL SAYAH**[1],
PATRICIA STOLF[1], **MINH-THUYEN THI**[1], AND **CHRISTOPHE VARNIER**[2]

[1]University of Toulouse, Toulouse 3 University, 31062 Toulouse, France
[2]Université Bourgogne Franche-Comté, UTBM, UFR ST, 25030 Besançon, France
[3]LAPLACE, CNRS, INPT, UPS, University of Toulouse, 31062 Toulouse, France
[4]EATON SAS, 38330 Montbonnot-Saint-Martin, France

Corresponding author: Jean-Marc Pierson (Jean-Marc.Pierson@irit.fr)

ABSTRACT As the need for cloud services has been growing steadily, the size and energy consumption of datacenters have increased significantly over the past years. Due to economic and environmental constraints, energy efficiency in datacenters and greenhouse emissions have become a major concern. Renewable energy is widely seen as a promising solution to supply datacenters using local energy, without greenhouse gas emissions. However, the intermittent power generation resulting from the use of renewable energy imposes a paradigm change in the way energy and computation activities are managed. On the one hand, service placement and scheduling may be used on the IT (information technologies) side to adapt to the available power. On the other hand, the storage units may be used to lessen power generation variations. Existing literature and actual deployment mainly design optimization algorithms including the entire system (from cloud service to electrical management, the latter often being neglected or simplified). Conversely to these approaches, we propose a solution where each side optimizes its own objectives, both interacting through a negotiation loop process to reach a common agreement. In this paper, we present DATAZERO, a project developing this idea to ensure high availability of IT services, avoiding unnecessary redundancies, under the constraints due to the intermittent nature of electrical and cloud services flows.

INDEX TERMS Cloud datacenters, renewable energies, optimization, middleware, negotiation, power models.

I. INTRODUCTION

Datacenters require a lot of energy to handle the IT load coming from the ever growing demand for online services. Several studies show that the energy demand on the IT industry is increasing on a large scale and at dramatic speed, in particular due to the datacenter industry [10], [11], [25], [56]. Today, datacenters rely on one or several power grids to supply the computing facilities with the demanding power. The electricity coming from these grids can however be more or less clean, in terms of CO_2 emission or nuclear waste. The development of renewable energies thus opens a way to power datacenters in a cleaner way. Innovative initiatives

make the shift from traditional dirty powered architectures to datacenters powered, at least partly, by renewable energy from solar panels (Aiso, Phoenix IT, Intel, Sun, Google, HP), from wind turbines (Google, OWC, Green House Data, Baryonyx), from geothermal and hydropower (Greenqloud), hydropower (Yahoo, Facebook) or from current generators based on fuel cells (Panasonic). To show their involvement towards more sustainable solutions, datacenter operators, like Apple and Google for instance, offload their production of green electricity to third party power providers, and pay the so-called RECS (Renewable Energy Certificate System) certificates. Doing this indirectly favors the production of renewable energy and operators can publicize their efforts in that direction. While worth mentioning, this solution is mainly a commercial trick since electrons always come from

The associate editor coordinating the review of this manuscript and approving it for publication was Arup Kumar Goswami.

the closest plants (might be coal). It also has several drawbacks compared to local production, such as the losses in transport and distribution (less than 10%).

Building a datacenter only fueled by renewable energy is however a challenge. Both the renewable energy production and IT load are intermittent, i.e., photo-voltaic panels are inefficient at night, wind turbines only produce electricity with wind and users submit their jobs when they want, but their peak periods do not necessarily coincide. It is therefore essential to store the energy to make them match. Batteries usually deployed in datacenters for handling short term incidents on the power line, are however not able to cover incidents lasting more than a couple of minutes or hours. Using electrolyzers to produce Hydrogen (H_2) from unconsumed power, storing it in tanks and using fuel cells to produce electricity again when needed is therefore complementary to batteries, wind turbines and photo-voltaic panels. Typically, batteries can be useful to store the extra power not consumed during the day to then be used during the night, while hydrogen storage helps to fill the power production gap between different seasons. Complex power systems must thus be implemented to ensure datacenter functioning.

Designing and operating a datacenter using renewable energy and long term storage is thus not trivial. In addition to choosing which type of power storage to use, there is a need to optimize the IT load in relation to the energy availability, and conversely, to optimize the energy production in relation to the incoming IT load. The sizing of the system in terms of IT and electrical components is also a challenge. Such optimization problems can be solved centrally, when the whole system is known, from the IT and electrical point of views. Such solutions have been subject to many works in the past, but suffer from the poor scalability and lack of dynamism when something changes on one of the sides. We propose an innovative approach where the IT part and the electrical part are able to handle and optimize their operations independently, but also to negotiate among themselves when necessary. For instance, when the IT load increases and more power is needed, or when a cloud covers the sun and the photo-voltaic production falls, each part may start to negotiate with the other part if it cannot handle the problem on its own. To the best of our knowledge, it is the first time that such a negotiation is introduced in this context to make the IT and energy constraints match.

In the DATAZERO project (2015-2019), the objective is to investigate the possible solutions to design and operate a datacenter in the best way, fueled only by renewable energy and that uses negotiation in case of unexpected events. In addition to the design of an efficient negotiation process, the objective also includes several aspects to get an holistic view of the problem: the efficient power control and power dispatching among different sources of energy, efficiently scheduling tasks on the IT servers, not forgetting the underlying problems to do with electrical and IT sizing.

The main contribution of our approach is to address the following issues and to group them in a common framework:

- propose a negotiation process giving hints to the optimization solvers (IT management and electrical management). During the negotiation phase, solutions for both electrical and IT sides are challenged and the best trade-off for each part is chosen.
- favor an approach where each side, electrical and IT, optimizes its own problem, instead of trying to solve a global optimization problem where one centralized algorithm knows all data (e.g., from electrical and IT parts, from short term to long term management). This way, both optimization problems can be solved using different methods, for instance with heuristics or linear programming, and independently, thus facilitating the use of better suited solutions in particular cases.
- include several renewable power sources and storage (solar panels, wind turbines, supercapacitors, H_2 fuel cells, batteries), modeling precisely their behavior with equipment testing and including aging effects in a comprehensive microgrid infrastructure.
- integrate the modeling of application profiles and server load profiles, allowing application execution at degraded performance, giving more flexibility when scheduling tasks, besides the classical effects on the IT equipment and jobs (switch on/off servers, DVFS (Dynamic Voltage Frequency Scaling), migration).
- propose an innovative middleware facilitating communication and negotiation between the electrical and the IT sides. The innovation lies in the fact that the middleware copes with both real hardware and simulation, or a mix of both (e.g., simulated electrical infrastructure and real IT servers, or vice-versa).

The typical datacenter targeted by the project is about 1MW peak of power demand, which is commonly encountered in enterprises and public datacenters, while the IT load is representative of Cloud services.

In this paper, some technical details are omitted due to lack of space, but the interested reader can refer to the public deliverables of theDATAZERO[1] project for further information.

The rest of the paper is organized as follows. The DATAZERO models, the electrical part's design, and the modeling of the constituting elements are given in Section II. Section III presents the different modules optimization problems: IT Task Scheduling (III-A), Power Decision (III-B) and Negotiation (III-C). Then, Section IV states the sizing problems for both IT and electrical sides. In Section V, we detail the developed middleware; Section VI outlines the role of real experiments used in conjunction with simulations. Finally, Section VII surveys the related works and projects, while Section VIII concludes the paper giving the next steps of the DATAZERO project.

II. THE DATAZERO MODELS

The DATAZERO project aims to conceive an innovative structure of a datacenter powered with renewable energy sources. Depending on the datacenter's size, the constraints,

---

[1] http://www.datazero.org

models and solutions may differ. As a first approach, we shall concentrate on datacenters installed on a ground area of around 1000 m^2 and that have an electric capacity of more or less 1 MW, cooling consumption included, distributed on two 500 kW branches that are used to provide N + 1 or 2N redundancy. The most disruptive constraint introduced is that we only use renewable energy produced locally within the datacenter.

In this section, we present the models defined and used in the project: the electrical architecture, with a particular focus on the redundancy issues, the energy production and storage, and at last the applications run on the IT resources. These models represent the characteristics of a whole datacenter's different components, including the power production. They are used in the next section as input of the optimization problems.

A. TOWARDS A 2N DATACENTER ELECTRICAL ARCHITECTURE

Redundancy is a main issue for datacenters when they must guarantee the quality of their service to clients. From an electrical point of view, redundancy is usually achieved by connecting the datacenter to several energy providers. According to [55], a ''2N'' datacenter means that the datacenter architecture comprises two complete power systems, each containing N components, run in parallel to hot swap between each other in case of failure. When using on-site and renewable energy, these definitions do not however have the same meaning, since the electrical supply does not rely on only one or two supplies but on several sources.

For this reason, we propose electrical architectures for datacenters meeting DATAZERO project constraints. One of their innovations is to extend the new concepts that have been designed in the GreenDataNet research project [24] to integrate renewable production. In particular, we use the GDN UPS (Green Data Net - Uninterruptible Power Supply) to secure the green supply. We propose two new architectures, named *Classical N+1* and *Breakthrough*, designed to power the datacenter's electrical loads (the IT components and the cooling) with respectively N + 1 and 2N redundancy. In the next paragraphs, we detail the hypotheses for each architecture. Advanced technical details on these architectures can be found in [8].

1) ARCHITECTURE AND HYPOTHESIS FOR A *CLASSICAL N + 1* DATACENTER ARCHITECTURE

The *Classical N+1* datacenter architecture intends to power, at a reasonable cost, the datacenter with renewable energy from Photovoltaic Panels (PV)/ Wind Turbines (WT)/ Fuel Cells (FC)/ Lithium-Ion batteries (Li-Ion)/ etc. as shown in Figure 1. In this architecture, one of the UPS, typically only integrating a battery, is replaced with an advanced GDN UPS, including renewable energy generation and hydrogen-energy storage. This UPS is in charge of supplying the IT resources with power or to distribute it to the storage elements if their is surplus. A connection to the grid is available as backup in
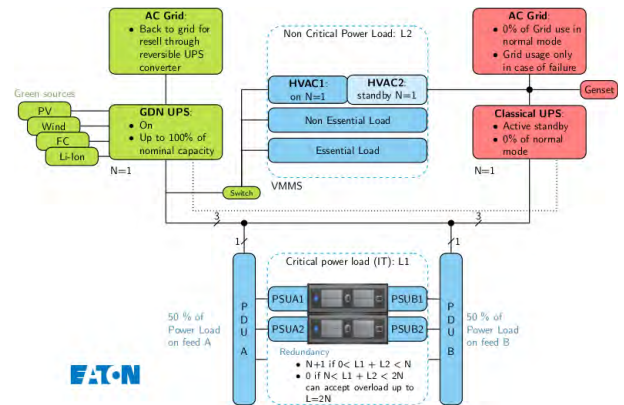
FIGURE 1. Classical datacenter architecture with N + 1 redundancy (Green + Grid).
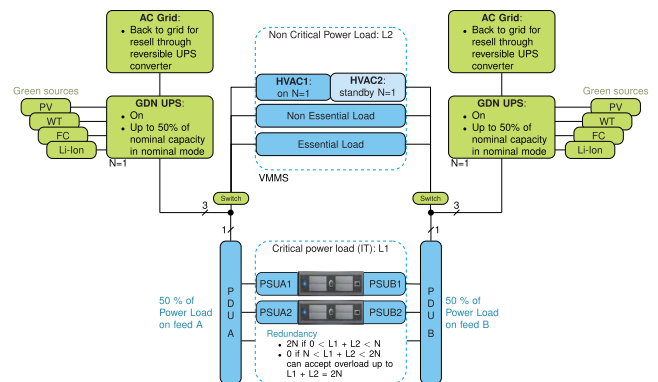
FIGURE 2. Breakthrough architecture with 2N redundancy (Green + Green).

case of local supply failure. Excess generation, if any, can also be sold to the grid, although this is currently not considered in the project. We set the following hypotheses for the design of this architecture:

1) The target datacenter redundancy is N+1: N+1 redundancy means that the number of components (generators, storage units and grid) provided is one more than the number (N) of components needed [55]. It is also considered as N+1 because the two power feeds are not fully separated (a fault could shut both paths down).
2) In normal operation mode: The datacenter only uses renewable power sources and does not access the Power Grid. Power Grid connection is considered for two purposes: to provide redundancy in case of failure and to allow power resell.
3) Critical electrical loads (racks and servers supporting the IT applications) are powered by two input branches (approximately 50/50%).

2) ARCHITECTURE AND HYPOTHESIS FOR *BREAKTHROUGH* DATACENTER ARCHITECTURE

The *Breakthrough* datacenter architecture is more ambitious, as shown in Figure 2, since only green generation is used to supply the electrical loads. It tunes hypotheses 1 and 2 from the previous *Classical N + 1* architecture, as shown below:

1) The target datacenter redundancy is now 2N. This means that the datacenter architecture comprises two complete systems, each containing N components, running in parallel. For instance, the GDN UPS are doubled, and hot swap between each one is feasible.

2) The power Grid is used in none of the modes (normal or failure), neither for normal nor for critical loads. The Grid connection is now optional and is only here to resell excess power.

The excess power generated by the green sources, i.e., the power not directly used by the loads, is preferably used to fill the batteries and to generate hydrogen. When the batteries and hydrogen tanks are full, the excess power can either be sold to the grid or through Hydrogen resell process. We do not however investigate further into reselling power in the remainder of the paper and leave this for future work.

3) INNOVATIVE ASPECTS OF THESE DATACENTER ARCHITECTURES

Considering "N + 1" or "2N" power architectures for datacenters is not novel, including when using renewable energies. However two aspects need attention: First, the GDN UPS used in DATAZERO is itself an innovation, a one of a kind equipment able to handle a peak load of 350kW, dispatching the power coming from PV and WT, to short term storage (Li-Ion) and long term storage (FC). It has been extended from previous works [24]. Second, closely studying the impact of such a novel equipment in terms of real platforms and simulation in a common framework (see Section V), and especially its link with the power dispatching decisions, is new.

B. ENERGY SOURCES AND STORAGE MODELING

In this part, we detail more precisely what are the electrical components used to power the datacenter, sources and storage, and how they are modeled.

To supply the servers and the cooling system, a dedicated power supply system is necessary. For both architectures described earlier, the green power sources (PV and WT) cannot reliably supply the power loads demanded by the IT infrastructure, as they lack of flexibility and controllability. Three complementary types of storage units are therefore used. Supercapacitors, with their low energy but high power, handle very short term fluctuations of power and ensure the stability of the system. Batteries, with a much higher energy capacity, enable to shift load or generation through time, but only over a few minutes or hours. Finally, to account for longer term variations and seasonal trends, hydrogen storage (combining an electrolyzer to generate hydrogen from electricity, hydrogen storage tanks, and fuel cells to generate electricity from hydrogen) is also used. These different power components are connected to the same DC (Direct Current) bus, where the loads are also connected. DC/DC and DC/AC (Alternating Current) converters are used to adjust the voltage of individual components to the DC bus voltage. Together, the obtained system is called a *microgrid* (Figure 3), i.e., a
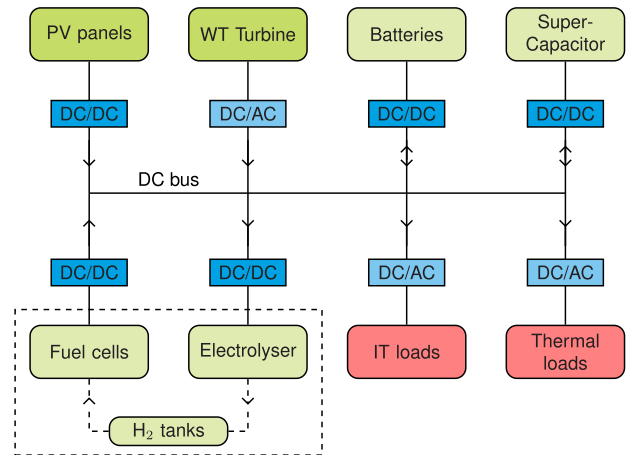
FIGURE 3. Structure of the DC microgrid supplying the datacenter loads.

small power system with local loads, generation and storage, that can operate isolated from the Power Grid.

To ensure reliable datacenter operation, the above components must be controlled adequately through energy management algorithms. As discussed in Sections III-B1 and III-B2, models are necessary to decide how much power each component should output, while meeting related operational constraints, such as the balance between generation and consumption, and the state-of-charge limits for batteries. Additionally, these models are used to simulate the electrical part of the system, and to validate the correct operation of the control algorithms before moving to physical experiments. In the following, we illustrate our work on models through the fuel cell example. Models for other electrical components are similarly described precisely in [8].

In practice, models define the characteristics and behavior of the components. For a fuel cell that converts hydrogen into electricity, the model describes the voltage of the fuel cell as a function of its output current which then enables to determine the corresponding hydrogen consumption. Figure 4 illustrates the behavior of our fuel cell model against measured results. It shows measured (continuous) and modeled (dashed) curves for a new, used and end of life fuel cell.

In reality, the performance of components and devices tends to degrade with time, for example due to materials' wear. For a fuel cell, the internal resistance tends to increase, so the voltage decreases. As a consequence, the maximum output current and power also decrease, and an aged fuel cell can only deliver a fraction of the output of a newer one. To account for this phenomenon, aging is considered in the proposed models, either using experimental data from participating laboratories or results from the literature. As shown in Figure 4, these data are used to derive prognostic models providing component characteristics as a function of their age.

In addition to enabling a realistic description of components, these prognostic models provide a way for the energy management algorithms and the negotiation to take decisions
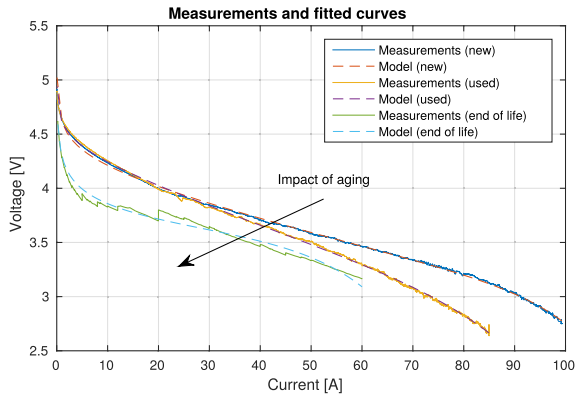
FIGURE 4. Fuel cell stack voltage curves from measurements run by FEMTO-ST/FCLAB and resulting model output for different aging levels (new, used and end of life).

that consider how the component performance will degrade, with an impact on how soon the component should be replaced. Such a factor is necessary as the marginal cost of the primary sources in the system is close to zero; in other words, wind and solar energy are free, and so is their storage. As a consequence, decisions should include capital (replacement) costs, and not only operation costs. An example is provided in (1) for the fuel cell. The fuel cell is assumed to be able to output a maximum amount of energy in its lifetime E_{fcMax}, and must then be replaced. The cost C_{fc} of operating the fuel cell at a given power P_{fc} and for a given duration Δt is therefore proportional to the capital cost C_{fcCap} of the fuel cell and the share of the lifetime energy (or energy throughput) consumed:

$$C_{fc}(P_{fc}, \Delta t) = C_{fcCap} \frac{P_{fc}\,\Delta t}{E_{fcMax}}. \tag{1}$$

Similar principles are used for modeling the other electrical components (PV panels, wind turbines, batteries, electrolyzer, hydrogen tanks, and supercapacitors) in the system. For each component type, several elements are connected in series and/or parallel to increase the output power and to form a microgrid that supplies the datacenter. For example, several batteries are used to reach the necessary voltage and storage capacity. The resulting models are then combined into a power system simulator that reproduces the behavior of the actual system, from initial data input (e.g., solar radiation) to DC bus voltage.

C. IT MODELS
The IT side encompasses the hardware infrastructure such as the servers organized in racks or the applications that are executed on these servers.

1) HARDWARE MODEL
In classical datacenters, each rack includes not only servers, but also support equipment such as networking or management systems. Studies [29] show that the consumption of this type of equipment is usually constant when powered on.

The model used in the DATAZERO project is then to have a constant power consumption for a switched on rack along with the power models of servers. The model for servers is the classical [17] linear model linking power consumption and computing load for a particular processor frequency. In most cases, the dynamic element of the power consumption of a server comes from the processor load.

The available levers to act on the IT power consumption are the ability to switch on/off a rack or a server, the ability to change the frequency of a processor and finally the ability to migrate an application from a server to another. For each of these levers, the models provide time and energy related information. As an example, the model for switching on and off a server provides the time needed to switch it on and off and the energy consumed during the switch.

2) APPLICATION MODELS
One of the objectives of a datacenter is to efficiently run a large number of applications. In DATAZERO, in order to evaluate the quality of the proposed solutions and to solve optimization problems, an accurate model of the applications is needed. Energy constraints are however an issue in the scheduling of these applications. Arrival of the applications and the computing resources allocated to them can be completely different from time to time, due to power availability for instance. For this reason, energy consumption characteristics of the applications must be included in the model. However, applications only consume power through their resource consumption. In our model, we consider that the energy consumed by an application depends on the resources it consumes. This model is also used to simulate the behavior of the resulting optimization processes.

Two models are used here: the *application* model intends to model the energy consumption of one application, while the *workload* model represents the arrival on the platform of a set of applications.

Applications executed on a datacenter depend on the context, and we focus in DATAZERO on Cloud applications. Cloud applications can range from High Performance Computing (HPC) applications to web services. We therefore distinguish between two kinds of applications: batch processes with a due date (HPC applications for instance), or services that never end and are expected to run at any time on the platform. A due date is the expected date for the process to end its execution. It is not a hard constraint: in case of overrun, the process is not killed and continues its execution. Due dates are used in QoS (Quality of Service) metric assessment.

Concerning application models, two aspects present an originality in DATAZERO compared to similar works:

Application Profiles: Usually, applications consume resources depending on their usage and on their workflow. For instance, a machine learning system usually alternates between learning and usage of the learned model. An application profile allows to describe the phases of an application. A phase [16] is a period of
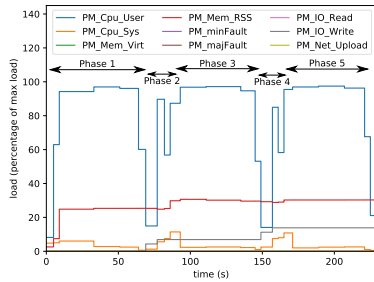
FIGURE 5. Profile of a raytracing application with five main phases. The first one is between 0 and 70s and is CPU-intensive, the second one is between 70 and 90s and is I/O-intensive. Then the phases come back alternatively.
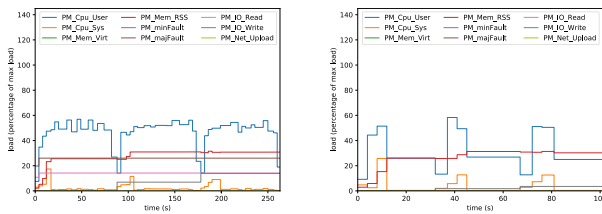
FIGURE 6. Profiles of a raytracing application with two different settings. The normal execution is on Figure 5. The profile resulting from the normal mode on a slower hardware is on the left (it lasts twice as long as the normal mode), the profile resulting from a degraded mode (i.e., requiring less computing power and leading to a lower quality of service) is at the right (it lasts less than the normal mode).

time where the application consumes a nearly constant amount of resources. As an example, Figure 5 shows an application with 5 phases where the resource consumption is nearly constant. The resource consumption in this example is related to CPU, memory and I/O consumption. This figure shows that some time is needed when resource consumption changes to detect a new phase.

Degradation levels: Several types of applications can adapt their execution depending on the context, the so-called elasticity. Under heavy load, they can work in a degraded level. A classical example is video streaming which can reduce the encoding quality. An application can then produce a set of profiles depending on the degradation level, which can be used by the IT decision center to execute the most relevant one depending on the context. An example of degraded profiles for a single application is shown on Figure 6. The three resulting profiles can be used, depending on the decision of the IT scheduling optimization. Note that a degraded execution can be obtained by dynamically scaling the frequency of the processors (DVFS) so that they consume less power, as in [63].

The presented model is able to represent both kinds of applications, from batch to services. The profile itself as shown on Figure 5 is described in [61] and is a XML description of the resources consumed over time. In DATAZERO, the XML descriptions of the profiles have been extended to take into account several degradation levels

for each application, to give more flexibility when dealing with variable power availability (see [8] for details).

In the context of DATAZERO, the application profiles of all tasks and the ones associated to each degradation levels will be provided. In an actual production system, this information is not available for not-yet executed applications. Similarly to [16], we plan to use learning methods to predict the profile of a running application as a function of its resource consumption profile during the first minutes of execution, as well as contextual data (information from user, type and amount of data).

3) WORKLOAD MODEL

To assess the quality of the decision and optimization algorithms, they must be tested against different set of applications, called workloads. A complete workload is composed of the set of applications, each with the following related information:

- Arrival time;
- Resource consumption profiles over time, using phases (described in II-C2): processor, memory, I/O, network, etc;
- Policy-related information: priority (representing that some applications are critical compared to others), due date (what is the deadline to execute an application), application-related QoS [26], [47].

A workload is defined by a SWF file (Standard Workload Format) following the standard described in [6] which gives the requirements, priority and arrival time for each arriving task or service. Along with this file, each task and service has a phase description file providing its resource consumption over time. Using these files, the IT scheduler decides when and where to execute each ready application in order to abide by the policy related information. Different policies will be used in the context of DATAZERO to test several scenarios.

D. MODELS OF THE NEGOTIATION SYSTEM

One of the original proposition of the DATAZERO project is to introduce a negotiation protocol to match the power consumption of the IT resources with the power production of the green sources, instead of doing a global optimization of the problem. The aim of this negotiation protocol is to introduce more dynamism when the initial conditions change. To implement this negotiation protocol, we propose to distribute the main roles of the power management in three modules, namely IT Decision Module (ITDM), Power Decision Module (PDM) and Negotiation Module (NM), as shown on Figure 7. These modules represent the core of the DATAZERO framework.

The modules cooperate by exchanging power profiles. A power profile is a set of power values for the different time steps of a given time horizon (the time interval used for negotiation). The concept of *power profiles* is used for representing, at any time during a given time horizon, either: (i) the power required for the IT part or (ii) the power proposed by the electrical part. Several profiles could be submitted for the
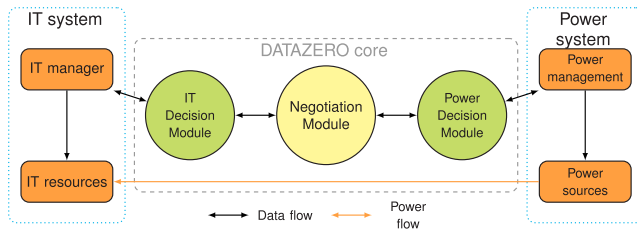
FIGURE 7. Negotiation architecture overview.

same time horizon. They can also be submitted for different time horizons that correspond to different time scales (from hours to days ahead negotiation).

The ITDM is in charge of managing the power needs of the IT resources. It uses the power profiles defined in the application model to compute a global power profile that is then used as proposition for the Negotiation Module. Conversely, the ITDM uses the profile negotiated with the Negotiation Module to decide which application to run, and which IT equipment to power on or off, depending on the QoS constraints.

The PDM is in charge of managing the power sources and storage. Based on the power system state, it proposes power profiles to the Negotiation Module. Once a profile is negotiated, the PDM is in charge of deciding which power source or storage to engage.

The NM is in charge of making the power profiles matching between the ITDM and the PDM. Its operation is further explained in Section III-C.

A utility value is associated to each profile. This value is computed by the corresponding DM (ITDM or PDM) at the time it proposes the profile. This utility is a metric that intends to value the quality of the profile for the DM. The utility represents how it fits the current state of the DM, the consumption forecasts and the confidence about these forecasts. This concept of "utility" is closely related to the one from Game Theory [50], representing the gain or loss from the point of view of the concerned Decision Module.

III. OPTIMIZATION PROBLEMS

In classical datacenters, only the IT resource scheduling is addressed. Even if energy saving is an important concern, the main objective remains the quality of service. In [54], Orgerie *et al.* have presented different ways to improve energy efficiency in datacenters. Mastelic *et al.* [49] have presented a survey on energy efficiency in datacenters. Sun *et al.* have designed a scheduling algorithm to map tasks while taking thermal constraints into account [60].

High availability in Cloud is allowed by a redundant provisioning of the resources, despite the energy consumption. Conversely, in DATAZERO, the energy is a crucial constraint since it is limited by the infrastructure of power sources and energy storage capacities. Consequently, IT resources are limited by the energy availability over time; IT scheduling algorithms have to take this constraint into account. Moreover, power-related decisions should consider

the IT load, the stored energy and the weather forecast, as well as short term and long term points of view, including aging. Thus, power and energy components have to be managed in order to satisfy the needed power for covering the IT load at best.

In the following, the underlying optimization problems to solve respectively the IT, energy and negotiation parts are proposed. While the different optimization problems could be solved by using either heuristics or exact methods, we decided to showcase both possibilities here, without loss of generality. For instance, the IT tasks scheduling problem is solved using heuristics, while a linear program is proposed for the electrical dispatching. The opposite could also have been exposed easily. Note that, since these optimization problems are based on estimated input values (weather forecast, expected load, ...) that are not always trustworthy on the long term, we only consider short time periods in the optimization.

A. IT SCHEDULING

The aim of IT scheduling is to allocate IT resources and schedule tasks in order to satisfy power and resource constraints while respecting quality of service. The schedule is made considering the infrastructure state (for example, the already scheduled tasks, the computing nodes (servers) that are powered on/off, the task profiles, etc.) and the availability of renewable energy, i.e., the power available during a given time horizon.

As already stated, two kinds of tasks are considered in DATAZERO: batch tasks and services. Batch tasks and services can be executed at different quality levels (degraded, where only part of the resources requested is delivered; or non degraded mode, where all the resources requested are delivered). The QoS for batch tasks is defined by the delay between the due date and the end of execution time (or the number of due date violations). For services, it is defined as the ratio between the resource percentage required and resource percentage obtained.

The IT scheduling algorithm will be called in different cases:

- when a set of tasks has to be scheduled. The algorithm computes different possible schedules, then sends the corresponding power profiles to the Negotiation Module (NM).
- when an event occurs on the power side, so that there is not enough power to execute the planned workload, the NM asks the IT Decision Module to recompute the workload.
- after agreement with the NM, the IT Decision Module aims to answer the following questions for each task: when is it possible to schedule it, where (on which node) and how (e.g., at which frequency)?

The results of the IT scheduling are either only one solution for the final decision or a set of solutions during negotiation, with QoS metrics associated to each of these solutions. These metrics are used by the Negotiation Module in the form of the utility presented in II-D in order to rank the profiles.

The IT scheduling problem in DATAZERO will be tackled using inputs, some constraints to be satisfied and an optimization objective. These inputs are:

- a power profile on a given horizon
- a set of tasks to be scheduled during a given time horizon
- the current state of the system (tasks already scheduled and in progress, status of physical and virtual machines)
- IT Infrastructure (Machines types and numbers, Performance-Power ratio, associated characteristics, green levers for On / Off, DVFS, Migrations: associated energy and time costs).

Two constraints will have to be satisfied: (i) the power consumption should be lower or equal to the power profile given as input; (ii) the mapping found should respect the resources availability (CPU, memory, machines, etc.).

The optimization objective is the maximization of the QoS, for example the minimization of the number of exceeding due dates. The mathematical formulation of this optimization problem has been done in [9].

The problem could then be solved with (i) greedy algorithms like Best Fit, presented in Algorithm 1, which tries to fit the tasks in the node having the smallest power consumption, respecting the power profile and resource constraints; or (ii) with meta heuristics like Genetic Algorithm [5] or linear programming (on small instances).

B. ENERGY MANAGEMENT

The Power Decision Module (PDM) is in charge of controlling the operation of the electric components within the system. It acts at two different time scales: the scheduling addresses long term scale, while the dispatching addresses short term. This led to two main optimization sub-problems:

1) Electrical Power Scheduling addresses the energy unit commitments. This optimization problem aims at defining, periodically and with a horizon of several hours to days, the power output of each component in the power system. Based on both negotiation steps with IT part (expected load) and weather forecast conditions, it defines how each component will be used over the selected horizon. The resulting schedule is especially useful for determining the best way to use storage units for an autonomous cycle usage of the datacenter, year after year. In fact, this schedule guarantees step by step the storage level such that overproduction periods compensate underproduction periods (daily and seasonal balance).

2) Electrical Power Dispatching addresses the power split between components. This optimization problem aims at defining, in real-time and at the timescale of the second, the current set point of each component in the power system in order to follow as much as possible the power output resulting from the PDM optimization problem. For example, the dispatching algorithm determines the output of each electrical component, from the renewable

---

Algorithm 1 Best Fit Constrained by a Power Profile Pseudocode

Data: Set of tasks in queue, set of available resources, power profile for the considered time horizon
Result: Tasks with assigned time and processor

1 sortTasks(queue)
2 slotsUsed ← dictionary⟨processor, timeIntervalArray⟩
3 **while** queue.hasTasks() **do**
4   t ← queue.getTask()
5   processor ← null
6   startStep ← t.release
7   **for** *currentP* in *availableResources* **do**
8   **if** haveEnoughPowerToSchedule(t, *currentP*,
9   powerProfile) **and** verifyConstraints(t, *currentP*,
10   slotsUsed, startStep) **then**
11   **if** (processor ≠ null
               and energyIncrease(*currentP*) <
               energyIncrease(processor)) **or** processor ==
               null **then**
12   processor ← *currentP*
13   **if** processor ≠ null **then**
14   schedule(t, processor, startStep)
15   updateSlots(processor, slotsUsed, startStep, t)
16   updateProfile(powerpProfile, processor, slotsUsed, startStep, t)
17   queue.remove(t);
18   **else**
19   startStep ← stepSize
20   stepSize++

---

sources to the storage units, depending on the amount of power decided by the negotiation process.

In the following, we detail these two sub-problems, using the notations defined in Table 1. Note that, for interested readers, a research report[2] more extensively details the linear programs. Also for readability reasons we have limited here the number of constraints and parameters, as the cut-in, cut-off values for the wind turbines, to keep the essential ones but they can be found in the report.

1) ELECTRICAL POWER SCHEDULING

Over a given time horizon, depending on weather conditions and the expected IT power load given by the Negotiation Module, the operations on the electrical components have to be adequately scheduled to guarantee a yearly autonomous usage of the datacenter. For example, if renewable energy production is expected to be very low on the next day, the storage units should be used to compensate the shortage. Thus, day after day, electrical power scheduling has to decide if storage devices charge or discharge, hour by hour, to make

---

[2]https://hal.archives-ouvertes.fr/hal-02081951

TABLE 1. Nomenclature.

Variable	Description
	Notations: Indices
k	Index for time step with scheduling horizon
K	Index of the last time step of the horizon
	Notations: Load
$Pload_k$	Load power $[kW]$
	Notations: Solar panel
Ppv_k	PV power $[kW]$
	Notations: Wind turbines
Pwt_k	WT power $[kW]$
	Notations: Batteries
$SOCmax$	State-of-Charge (SOC) upper limit $[kWh]$
$SOCmin$	SOC lower limit $[kWh]$
η_{ch}	Battery charge efficiency $[\%]$
η_{dch}	Battery discharge efficiency $[\%]$
σ	Battery self-discharge rate $[\%]$
SOC_k	SOC at instant $k\Delta t$ $[kWh]$
Pch_k	Power used to recharge battery $[kW]$
$Pdch_k$	Power discharged from battery $[kW]$
	Notations: Hydrogen tank
$LOHtarget_d$	H_2 tank level targeted at the end of the horizon
$LOHmax$	H_2 tank upper limit $[\%]$
$LHVh_2$	H_2 lower heating value $[kWh.kg^{-1}]$
$HHVh_2$	H_2 higher heating value $[kWh.kg^{-1}]$
η_{tank}	H_2 tank efficiency $[\%]$
LOH_k	H_2 tank inventory level $[kg]$
	Notations: Electrolyzer
$Pezmax$	Electrolyzer power upper limit $[\%]$
$Pezmin$	Electrolyzer power lower limit $[\%]$
η_{ez}	Electrolyzer efficiency $[\%]$
Pez_k	Power put into electrolyzer $[kW]$
Qez_k	Electrolyzer H_2 mass $[kg]$
	Notations: Fuel Cell
$Pfcmin$	Minimum power to operate fuel cell $[kW]$
$Pfcmax$	Maximum power delivered by fuel cell $[kW]$
η_{fc}	Fuel cell efficiency $[\%]$
Pfc_k	Power delivered by fuel cell at period k $[kW]$
Qfc_k	Fuel cell H_2 mass $[kg]$
	Notations: Inverter
η_{inv}	Inverter efficiency $[\%]$

this compensation mechanism possible. Moreover, this daily compensation mechanism has to be expanded to a horizon of one year because of seasonal variations. The optimization problem for a given time horizon addresses this electrical component management that guarantees the negotiated IT power load, such that the daily and the yearly compensation is maintained.

This type of scheduling can be formulated as a rolling-horizon optimization problem, where the objective is to manage the long term storage as close as possible to the expected level designed by the sizing study of the platform. An ideal level of Hydrogen at the end of each scheduling horizon is defined all over the year. The daily compensation is added as a constraint of the optimization problem, along with the ones that enforce the satisfaction of the IT power demand. One can give a formulation of this problem as the following mathematical program, considering these hypothesis:

- The time horizon, of duration T units of time, is discretized into K Δt small periods, each of them corresponding to one index $k \in [\![0, K-1]\!]$;
- The ideal level $LOHtarget_d$ of hydrogen is known all over the year, and precisely at the end of the horizon

$t = d\Delta t$ from the beginning of the year ($0 \le d \le D/\Delta t$ with D the duration of one year in units of time);

- The power comes first from renewable sources (PV and WT) and is directly used by the datacenter or stored either in batteries (Li-Ion) as storage elements (SE) or in hydrogen form, thanks to electrolyzer devices (EZ); if the power from renewable sources is greater than the power demand, the overproduction is stored into short term (batteries) and long term (hydrogen tanks) storage devices;
- The power Pfc_k delivered at each time step k by the fuel cell is bounded by $Pfcmin$ and $Pfcmax$; if the fuel cell is in use at time step k, its power value is proportional to the hydrogen quantity Qfc_k that is consumed during that period of time (Equations (4) and (11));
- The power Pez_k delivered at each time step k by the electrolyzer is bounded between $Pezmin$ and $Pezmax$; if the electrolyzer is in use at time step k, its power value is proportional to the hydrogen quantity Qez_k that is produced during that period of time (Equations (5) and (12));
- The state of charge SOC_k of the batteries at time step k depends on the previous time step, the auto discharge rate σ and the power of charge Pch_k or discharge $Pdch_k$ (Equations (6) and (7));
- The level of hydrogen LOH_k stored at time step k is bounded between 0 and $LOHmax$ and depends on the hydrogen consumption (Equations (8) and (14));
- The level of hydrogen stored as LOH_K at the end of the horizon T should be as close as possible to the target $LOHtarget_d$ to maintain the sustainability of the datacenter (Equation (2));
- The batteries only assure the daily smoothing. Then, the state of charge of the battery should be the same after each 24 h run (Equations (9) and (10));

To define the power scheduling during time horizon T, the following mathematical problem has to be solved to compute the variable values that minimize the difference between the targeted ideal level of hydrogen ($LOHtarget_d$) and the one (LOH_K) obtained at the end of the horizon (Equation (2)):

$$\text{Minimize} \quad |LOHtarget_d - LOH_K| \tag{2}$$

Subject to: $\forall k \in [\![0, K-1]\!]$

$$Pload_k \le Pwt_k + Ppv_k + (Pfc_k + Pdch_k)\eta_{inv} - (Pez_k + Pch_k)\eta_{inv} \tag{3}$$

$$Pfc_k\Delta t = LHVh_2 \times Qfc_k \times \eta_{fc} \tag{4}$$

$$Pez_k\Delta t = HHVh_2 \times Qez_k/\eta_{ez} \tag{5}$$

$\forall k \in [\![0, K]\!]$ and if $Pch_{k-1} > 0$

$$SOC_k = \min\{(1-\sigma)SOC_{k-1} + \eta_{ch}Pch_{k-1}\Delta t, SOCmax\} \tag{6}$$

$\forall k \in [\![0, K]\!]$ and if $Pdch_{k-1} > 0$

$$SOC_k = \max\{(1-\sigma)SOC_{k-1} - \frac{Pdch_{k-1}}{\eta_{dch}}\Delta t, SOCmin\} \tag{7}$$

$\forall k \in [\![0, K]\!]$

$$LOH_k = LOH_{k-1} + Qez_{k-1} - Qfc_{k-1}/\eta_{tank} \tag{8}$$

$$SOC_K = SOC_0 \quad \text{if} \ K < 24 \tag{9}$$

$$SOC_k = SOC_0 \quad \text{if} \ K \geq 24 \ \text{and} \ k\%24 = 0 \tag{10}$$

Bounds: $Pfcmin \leq Pfc_k \leq Pfcmax \quad \forall k \in [\![0, K-1]\!]$ (11)

$Pezmin \leq Pez_k \leq Pezmax \quad \forall k \in [\![0, K-1]\!]$ (12)

$SOCmin \leq SOC_k \leq SOCmax \quad \forall k \in [\![0, K]\!]$ (13)

$0 \leq LOH_k \leq LOHmax \quad \forall k \in [\![0, K]\!]$ (14)

Initial conditions: $LOH_0 = LOHinit$

$$SOC_0 = SOCinit$$

The mathematical program has been linearized [28] to be solved using an Integer Linear Program solver such as Gurobi [31]. The result of such a commitment is a power profile over time that each component of the power system has to respect step by step. However, as there is significant uncertainty on input data (e.g., on renewable generation and IT load), the scheduling algorithm has to be able to periodically propose new solutions that take new constraints into account. To define an appropriate time window we have tested several configurations with different real-case traces with windows of 24, 72, 168 hours. Our results show that a 72 time window allows the best decisions since the 24 hour one generates too much variations while the 168 one use inaccurate forecasts.

This formulation is a first approach of the electrical power commitment that minimizes the difference between a level of hydrogen target and the level of the hydrogen computed at the end of the period to ensure the seasonal offset. In order to obtain consistent solutions, mutual exclusion constraints have been added before the linearization step to guarantee a normal usage of the storage devices (e.g., charge/discharge of the batteries, fuel cell vs electrolyzer).

Note that, as the decision process uses a 72 hour time window, the time period is too short to have a noticeable change in SOH (State of Health), linked with the aging, of the components. We therefore do not take it into account. On the other hand such SOH could easily be introduced by changing the characteristics of the components, e.g. $SOCmax$ and $SOCmin$ for the batteries, between two computations.

If the PDM does not find any solution that respects all the problem constraints because of uncertainty on input data, a new negotiation has to be requested to converge to a power profile demand that is compliant with the power component states. In this case, the Power Decision Module receives new profiles or proposes alternative power profiles for which the distance – in term of power to deliver – with the current profile is as small as possible, and it exhibits new power component commitments. When a new profile is chosen at the end of a negotiation process, a new electrical power scheduling optimization problem has to be solved: (i) for the next time horizon or (ii) for the period of time that ends when external events make the current solution obsolete (e.g., unexpected changes in power production due to a passing cloud on PV, electrical problem of one power component, etc.).

2) ELECTRICAL POWER DISPATCHING

As introduced above, several energy sources and storage units are used to supply the datacenter. This implies that each component must be accurately controlled to respect its own operation constraints (e.g., state-of-charge limits of batteries) as well as the system constraints (e.g., power generation must be equal to power consumption at any time). This is achieved by the dispatching algorithm of the PDM, for which the main goal is to ensure the electrical stability and security of the system.

Due to its very short operation time frame, the algorithm does not use any prediction, and is thus purely reactive. Dispatching is also required to compensate for the errors arising from the schedules established earlier by the electrical commitment algorithms. For example, due to uncertainty on PV generation and on IT load, the preliminary schedule may be slightly erroneous, which means that the dispatching algorithm has to compensate the forecasting errors by changing the output of some components (e.g., by increasing the battery discharge power).

A simplified formulation of this problem is given below for a given value of pulse $k > 0$, which represents the time rate at which a new dispatching has to be done, depending on the previous state:

Minimize $|Pload_u - (Pwt_u + Ppv_u + (Pfc_u + Pdch_u)\eta_{inv}$

$$-(Pez_u + Pch_u)\eta_{inv}| \tag{15}$$

Subject to: $Pload_u \leq Pwt_u + Ppv_u + (Pfc_u + Pdch_u)\eta_{inv}$

$$-(Pez_u + Pch_u)\eta_{inv} \tag{16}$$

$$SOC_u = (1 - \sigma)SOC_{u-1} - \frac{Pdch_{u-1}}{\eta_{dch}}\Delta t$$

$$+\eta_{dch}Pch_{u-1}\Delta t \tag{17}$$

$$Pfc_u\Delta t = LHVh_2 \times Qfc_u \times \eta_{fc} \tag{18}$$

$$Pez_u\Delta t = HHVh_2 \times Qez_u/\eta_{ez} \tag{19}$$

$$LOH_u = LOH_{u-1} + Qez_{u-1} - Qfc_{u-1}/\eta_{tank} \tag{20}$$

Bounds: $Pfc_u \leq Pfcmax$ (21)

$Pezmin \leq Pez_u \leq Pezmax$ (22)

$SOCmin \leq SOC_u \leq SOCmax$ (23)

$0 \leq LOH_u \leq LOHmax$ (24)

$Pfc_u \geq 0$

$Pez_u \geq 0$

Initial conditions: $LOH_0 = LOHinit$

$$SOC_0 = SOCinit$$

$$Pfc_0 = 0$$

$$Pez_0 = 0$$

In the above formulation, the objective is to minimize the difference between the power load and the total power from the sources and storage units, i.e., minimize the power imbalance (Equation (15)). The constraint given in Equation (16) is that the total generation and output of storage units should be at least equal to the load. The Equation (17) updates the state-of-charge of the battery, while the Equation (18) updates the level of hydrogen in the tanks. The bounds are similar to those used for the scheduling problem formulation (Equations (21)–(24)).

This formulation is very similar to the electrical power scheduling formulation presented above, except that the scheduling fixes K values for each mentioned variable along the whole time horizon $K\Delta t$. For all $k \in [\![0, K - 1]\!]$, the dispatching optimization considers a time horizon Δt discretized in n pulses of length δt time units. For a given value of k and any pulse $u \in [\![0, n - 1]\!]$ ($k\Delta t \leq u\delta t < (k + 1)\Delta t$), the value of variables obtained by solving the scheduling problem are the setpoint values of the target controller. To be efficiently solved in real time, the mathematical program has first been linearized.

This control in real time allows for delivering the power load reference that the energy components have to provide. Considering all available sources and some natural uncertainties (weather, faults, incidents, etc.), an efficient reflex-dispatching has to be implemented. For example, a set of expert rules (e.g., use the battery in priority) or a Fuzzy Logic-based control in [51] is implemented to locally change the power dispatching between sources impacted by an unexpected (non predicted) event.

C. NEGOTIATION BETWEEN IT AND POWER SYSTEM

The goal of this optimization problem is to find an agreement between IT power consumption and electrical production. The Negotiation Module aims at guiding both decision optimization modules ITDM and PDM in order to converge to a power profile: (i) close to the one requested by IT and (ii) feasible by the electrical system. Without constraints, IT Decision Module prefers to execute jobs when it is better for QoS. However, this is unlikely to match with the best usage of renewable energy and storage devices. It is mandatory to compare the power coming from the execution of IT requests with the energy availability computed by the PDM. The Negotiation Module objective is to clinch the deal between the needs and constraints of the two Decision Modules (DMs).

The negotiation does not aim at solving a global optimization, but rather to find a good trade-off with as little specific information as possible. The detailed models of the IT and electrical infrastructures and their internal constraints are not known by the Negotiation Module. To find a compromise, the negotiation process only uses a few metrics associated to each power profile sent by the decision modules (e.g., the utility metric).

Figure 8 gives an overview of the information exchanged during the negotiation process. A complete negotiation
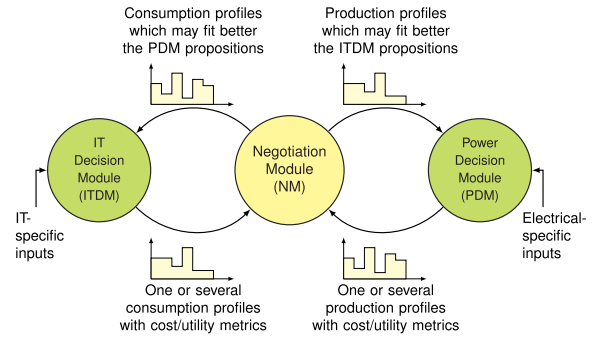

FIGURE 8. Illustration of the negotiation process, with the information exchanged between the Decision Modules and the Negotiation Module.

is usually completed through multiple successive *rounds*. At each round, the Negotiation Module may send a set of "hint profiles" to the ITDM or PDM for the next round. Those hint profiles are used to guide each Decision Module to some parts of the solution space considered attractive by the Negotiation Module. Based on their own information and on the previous hint profiles, the Decision Modules send new power profiles to the Negotiation Module. Doing so, the profiles are refined step by step from each side in order to explore the possible solutions and to find a reasonable compromise.

The negotiation begins at the initiative of either ITDM (new tasks submitted, tasks finishing, etc.), PDM (changes in the production forecast, failing element, etc.) or Negotiation Module (NM) itself (to guarantee continuity in case a long time is spent without negotiation).

The Negotiation Module implements the following steps to find an agreement: First, if one of the DM (ITDM or PDM) starts the negotiation process, the Negotiation Module receives a negotiation request, containing power profiles, each with its utility metric. If instead the negotiation starts at the Negotiation Module's initiative, the NM generates an initial set of hint profiles based on the results of previous negotiations. Until a satisfying solution is found or the maximum number of iterations is reached, the Negotiation Module sends hint profiles to both DMs, waits for their reply with counter-proposals and generates next hint profiles. To obtain these next hint profiles, the NM tries to match previously received power profiles of both DMs.

Different matching algorithms could be used, each one leading to different behaviors in terms of exploration and convergence. For instance, the Negotiation Module could identify the time steps for which the profiles diverge and generate accurate hint profiles taking account the amount of power that should be either restricted or increased.

Using a game theory approach, the Negotiation Module selects the best combinations of profiles to continue the negotiation.

As soon as a profile that enables reaching an agreement with both DMs is found, considering that the utility is high enough for both, the negotiation ends. In that case,

a reasonable compromise has been found. The selected profile is then sent to the PDM and ITDM, informing them about which profile has to be implemented for the concerned time horizon. However, if no compromise is found after a certain number of iterations, the negotiation phase ends anyway. A profile is then chosen from the ones sent by the PDM, in order to implement it (therefore guaranteeing that a decision is made). A new negotiation is started for another time horizon when a new event occurs, even if the previous time horizon is not completed yet.

IV. INFRASTRUCTURE SIZING

Defining the datacenter infrastructure implies to define how many IT resources, power sources and storage units are needed. Electrical and IT sizing problems aim at defining how to build the datacenter for the electrical and the IT parts. These sizing problems consider the most challenging scenario of a datacenter that does not use the Power Grid, called *breakthrough infrastructure* in II-A2.

Finding the most appropriate infrastructure of the whole datacenter is a complex optimization task that links both IT and Electrical sizing. A modeling that is able to join the two sizing problems in one optimization problem is probably the right approach. Moreover, if the usage of the datacenter depends on the infrastructure, the infrastructure also depends on the usage of the datacenter. An additional iterative approach has to be carried out during this sizing task, taking the interaction with the usage into account. At the same time, a certain level of robustness will be introduced within this optimization process to make the datacenter as efficient as possible, even if usage conditions are changing to a given extent. One option is to consider pessimistic scenarios that imply to slightly oversize the long term storage to handle at the same time bad weather conditions and an increase of the computation load.

The principle of an electrical sizing is the following. If the datacenter aims to be autonomous in terms of energy production, the totality of the energy comes from primary sources. The specification is to deliver twice a power of 500kW within a 2N infrastructure, meaning that the sizing has to be able to overcome a large power shortage.

As the primary sources (PV and WT) are intermittent, the lack of energy production in winter must be balanced by an overproduction in summer. This overproduction is converted into Hydrogen (H_2) to constitute the long term storage. The H_2 is stored into low pressure tanks. Given the efficiency of this reversible process and the energy production difference between days, it is possible to know: (i) how much energy is produced by the primary sources, (ii) the amount of energy that has to be stored. It is also possible to know how many H_2 tanks are needed. However, an optimization stage is necessary to know where the primary energy comes from (PV or WT). This depends on different objectives that have to be exhibited to measure their impact on the sizing. This sizing has to be completed with a short term energy storage that balances primary energy production fluctuation
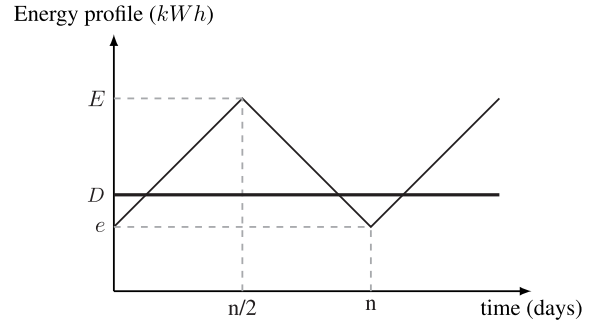
FIGURE 9. Primary energy produced by PV and WT.

between nights and days. These short-term storage devices consist of batteries whose number will be determined by this fluctuation. Similarly, very short-time disturbances will be controlled using supercapacitors close to each primary source and their power electronics.

As an illustration of our approach, we propose the sizing of the WT and PV platform in the particular case given in [27]. In this case, we assume that the wind speed distribution can be represented as a typical curve of the photovoltaic system [48]. The production of primary source's energy during one year linearly increases when days get longer and linearly decreases when days get shorter (see Figure 9).

As a result, it is obvious to find the best energy production day $n/2$ (with an energy production of E) and the worst one n (with a production of e) as shown in Figure 10. Let D be the energy demand for one day of the data-center. If we consider that D has a constant value over each day of the year, considering our model, the overproduction of each day has to offset the underproduction of another day. And using both wind turbine and photo-voltaic panel models, we can estimate their electrical production in kWh during these two days. This allows us to evaluate the needed energy in a first step and the primary source sizing in a second step. The energy production E can be expressed as:

$$E = \frac{D \times (1 + \eta)}{(\eta + \alpha)},$$

where η is the efficiency ratio of the long term storage (electrolyzer and fuel cell) and α is the proportional coefficient between E and e ($E = \alpha e$).

For instance, for the particular case given in [27], we found that the best configuration was to use only 1 wind turbine (500 kW) and 4787 m^2 of solar panel to meet the needs of a workload which consumption varies between 100 and 500 kW.

Taking into account the storage efficiency, the model introduced is not accurate anymore. The overproduction part needs to be larger than the lacking production part, as it is multiplied by the storage efficiency. The real model representing the system can be defined in Figure 11.

The sizing of the hydrogen system consists of computing the area of the triangle above the demand as in (25), in order
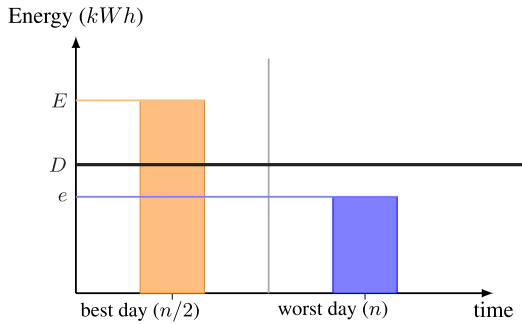
FIGURE 10. Primary energy produced by PV and WT, respectively for the best and the worst day.
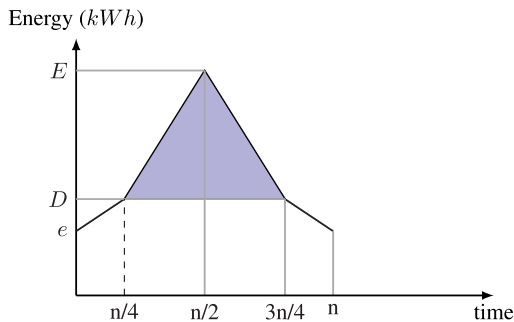

FIGURE 11. Required overproduction (gray part) for long term energy storage that takes efficiency ratio into account.

to get the energy to be provided:

$$E_{H2} = \frac{\frac{D \times 24 \times (1+\eta)}{\eta+\alpha} \times \frac{n}{2}}{2}. \tag{25}$$

Finally, this energy is converted into a volume of hydrogen to get the number of hydrogen tanks needed to store enough energy for the underproduction period.

At the same time, the IT sizing has to evaluate the accurate IT infrastructure that: (i) is the most efficient in terms of computation; and (ii) does not consume more than the power profile defined by the Electrical part. The link with the Electrical problem sizing is that choosing the IT infrastructure has an impact on this profile, on the way the power demand is able to change and on the dynamics of these changes. This is the reason why it is mandatory to build an unique model for the infrastructure sizing.

V. MIDDLEWARE

All the previously presented concepts are implemented in the DATAZERO platform, a testbed designed to validate our propositions. The objective of this platform is to support interactions between the IT environment and the power management that controls the renewable power sources. The platform is composed of three main parts, the IT System, the Negotiation System and the Power System, together with the Information System that traces all the activity in the platform and feeds the GUI System. Figure 12 gives a global view of the DATAZERO platform architecture with its main parts:
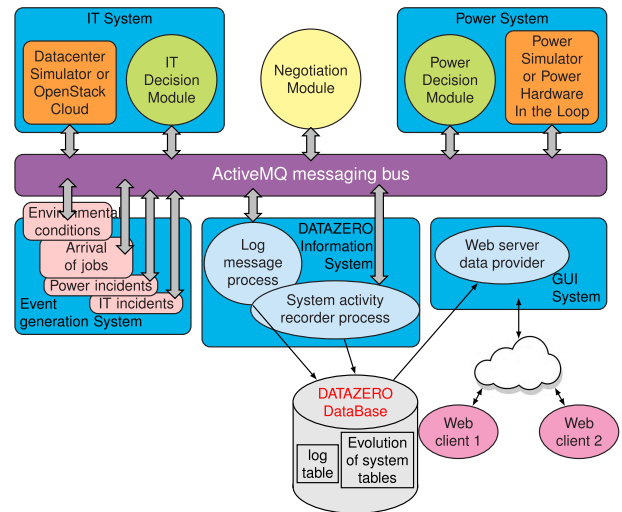

FIGURE 12. Global view of the DATAZERO architecture.

- The IT System is composed of the datacenter manager that monitors the datacenter (it can be either simulated with DCWoRMS [37], SimGrid [4] or real as an OpenStack [53] environment) and the IT Decision Module (ITDM) that interacts with both the datacenter resource manager and the Negotiation Module (NM). It takes decisions concerning IT resources allocation.
- The Power System is composed of the power components (simulated, emulated or real, as described in Section VI) and the Power Decision Module (PDM) that interacts with both the power components and the Negotiation Module to take decisions concerning power components operations.
- The Negotiation Module (NM) that interacts with both the ITDM and the PDM to arbitrate between the IT power demand and the availability of power that the Electrical side is able to produce.
- DataZero Information System has two main objectives. It stores all messages sent at system level, allowing post-analysis, replay sequences and extraction of specific data for other module usage. It also stores system activities in structured tables, to easily extract information for a Graphical User Interface (GUI). The Information System uses the MariaDB database.
- The GUI is based on a Web interface built to monitor several parameters in the overall datacenter. It is developed with the Angular framework, using "Web Server data provider" to retrieve data via HTTP requests. This graphical interface may serve as a control console to tune the policies of decision modules in the future.

By distributing the management of the whole platform into distinct modules, rather than grouping all the decisions in only one module, we enforce the possible development and/or deployment of dedicated policies, and particularly dedicated optimization algorithms as previously presented.

These modules interact with other external information providers to gather information used in their decision processes, e.g., weather forecast and other environmental conditions, workload, etc. A monitoring tool is implemented within the platform to give a global view of the system dynamics through the GUI. This monitoring tool gathers all the information available in the system and aggregates them into pertinent indicators.

As the modules may not be run on the same hosts, their interactions rely on network communications. To connect the modules, we use a middleware layer that supports these communications. Communications between modules are usually asynchronous and more on a peer-to-peer scheme than on a client-server mode. For that reason, we choose a Message Oriented Middleware (MOM) to facilitate the information distribution inside the platform. MOMs usually allow several communication modes as: (i) peer-to-peer, based on a message queue; or (ii) event-oriented, based on a topic object on which publishers can broadcast an event message to the subscribers of the topic. Moreover, as an intermediate between the communication protagonists, MOMs can implement additional services such as transaction management or exchange recording (persistence). For the DATAZERO platform, we use the ActiveMQ[3] MOM.

On this message bus, several topics are defined to structure the exchanged information, one for each type of event triggered in the platform. Every module willing to access an information is subscribing to the corresponding topic. We choose to allow a large access to information inside the platform and every event is exposed through a topic to all modules connected to the bus. More than 35 topics are defined in the platform and a new one can easily be added to guarantee the platform evolution. For instance, the *IT_MACHINE_CHANGE_STATE* topic is used to send and receive machine switch on events.

ActiveMQ supports several communication protocols at low level, such as OpenWire, AMQP or Stomp. This facilitates its use in heterogeneous environments or the addition of new modules. ActiveMQ also supports several programming languages such as Java, C++ or Python. Depending on the module, the development is done in one of these three languages and, for that reason, the low level protocol used in the platform is AMQP. Moreover, to facilitate the communication between heterogeneous components, possibly written in different languages, we choose to encode the messages in JSON (JavaScript Object Notation), a lightweight data-interchange format. In modules written in Java, we use the Gson[4] library to translate the Java objects used in the modules into JSON. For each event in the platform, a message type is defined and its JSON scheme generated. For instance, when a machine changes its state, a `machineChangeStateMessageContents` is

[3]http://activemq.apache.org/
[4]https://github.com/google/gson

```
"machineChangeStateMessageContents" : {
  "type" : "object",
  "id" : "urn:jsonschema:message:
              MachineChangeStateMessageContents",
  "properties" : {
    "machineId" : {
      "type" : "string"
    },
    "newState" : {
      "type" : "string",
      "enum" : [ "MACHINE_OFF", "MACHINE_ON",
              "MACHINE_START", "MACHINE_SHUTDOWN",
              "MACHINE_UNREACHABLE" ]
    }
  }
},
```

FIGURE 13. Code: Excerpt of the JSON format for the `machineChangeStateMessageContents`.
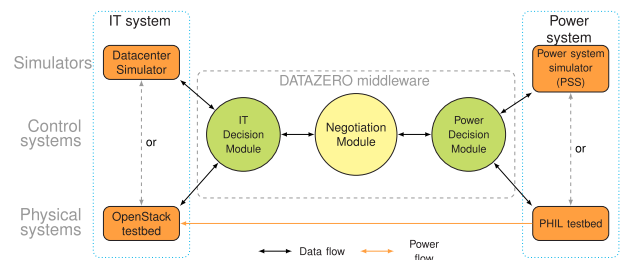

FIGURE 14. System overview, including simulators (datacenter simulator and power system simulator) and physical parts (IT OpenStack and PHIL).

generated. The following code (Figure 13) gives the JSON format of this message.

## VI. REAL HARDWARE, SIMULATIONS

In order to test and validate the correct operation of the system, simulations and experiments are required. First, simulations are needed to validate component models, and make sure they adequately describe their behavior accurately. When necessary, parameter tuning can also be done based on actual experiment results taking into account measurement bias [13]. Simulations are also useful to validate the correct operation of the control and energy management algorithms, e.g., with respect to the balance between supply and demand.

Then, as explained in Section II-B, models are simplified and approximated representations of actual, physical components, so experiments are also needed. Such experiments are designed to verify whether the simulations results are valid, or if adjustments in the models or control strategies are required (for example, if a phenomenon that was neglected should not be). As a consequence, electrical and IT optimization will be validated for some parts with simulations and some others with real hardware. An overview of the system is shown in Figure 14, where both simulation and experimental parts are shown, for both the power and IT sides.

To ensure a correct synchronization of the simulated parts, IT and electrical, a metronome is introduced. This metronome is a server in charge of giving a common tick to the different simulators so that they synchronize their virtual time and run at the same speed. The metronome is needed because both sides use different simulation environments that may run at
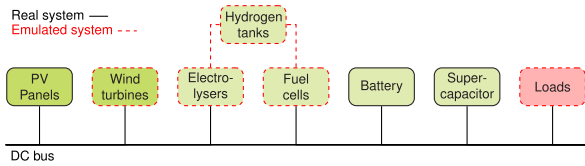
FIGURE 15. Proposed PHIL testbed for experimental validation of the models and control strategies.

different speeds. Moreover, the metronome is also necessary to allow to mix simulated parts and real control in the same experiment. Remember that the IT or the Power system can be either real or simulated, complexifying their synchronization. In that case, the metronome gives a reference time to the simulated part so that it stays at the right time step, following the real platform. Finally, the metronome is not used when both parts run on actual hardware without simulation.

A. ELECTRICAL SIDE
As a first step, a power system simulator is created in Matlab in order to simulate the behavior of the microgrid and its interactions with the other modules, such as the PDM. This simulator includes the electrical components' models and enables a preliminary validation of the control and energy management strategies, e.g., for dispatching and scheduling.

The next step would be a physical, full scale experiment. However, such demonstrations are costly, and expenses can easily amount to several million euros for a MW-scale system. A small scale experiment is thus used, where some components are emulated or only simulated.

To do so, a concept called Power Hardware-In-the-Loop (PHIL) is used. PHIL combines virtual and physical elements into a single system. The size of physical components is also usually smaller (e.g., at the kW scale instead of the MW scale), which implies significant savings with limited loss of representativeness. For example, the test-bed can include: i) a real photo-voltaic power plant and batteries, ii) an electronic load to represent the datacenter load (and absorb the output of the above components), and iii) real-time simulated-models for the rest of the system, such as for the hydrogen storage system. Another advantage of such a setup is the ability to run critical tests that could not be achieved in reality, such as short circuits. An example of proposed test setup is shown in Figure 15.

In order to be integrated with the IT side, the PHIL testbed is interfaced with the middleware. Messages including measurements, set points, and profiles are exchanged between the different components, the PDM and the rest of the system. It is also expected that the entire system should not know whether the microgrid is simulated, partially emulated with PHIL, or entirely real. By extension, the electrical part and IT part can be run in two different locations, without much impact on the results.

B. IT SIDE
For the IT side, we also consider the two different approaches, through an IT simulator and using a real infrastructure. Here,
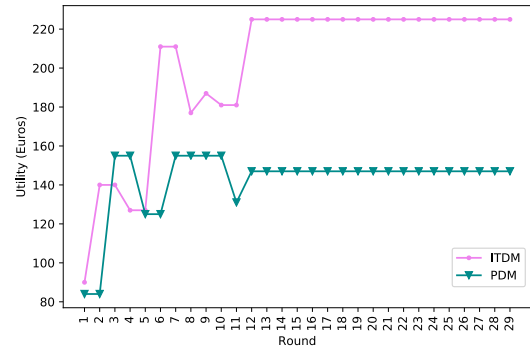
FIGURE 16. Utility of both ITDM and PDM over the several rounds of negotiation.

the objective is to be able to simply replace the simulator by the real infrastructure without any modification of the IT Decision Module. For this reason, the simulator needs to provide the same abstraction as the real platform and the same communication protocol.

For the simulated environment, we use SimGrid [4] which is a distributed system simulator. It aims to evaluate allocation and scheduling policies for a given workload on a given infrastructure. Different metrics are computed and green levers such as DVFS [30] or task migration can be simulated. For the real platform, we use OpenStack,[5] a cloud operating system that allows us to control compute, storage, and networking resources of datacenter through an API.

The communication between the IT Decision Module and the two different environments (real or simulated) is done using the middleware presented in Section V and "plugins" on Simgrid and OpenStack. When the ITDM sends messages that concern the IT infrastructure, they are interpreted by these plugins that: (i) translate the messages received to known commands in Simgrid or (ii) calls specific functions in the OpenStack API, when considering the real platform.

C. PRELIMINARY NEGOTIATION RESULTS
1) CONVERGENCE
As a proof of concept, in Figures 16 and 17, we present the preliminary results of the negotiation algorithm in an integration with different modules. The objective is to find a compromise in power profile while respecting the utility of both ITDM and PDM. In this experiment, we use real data, i.e., the IT workload generator comes from [12] and the weather conditions used to compute the power profile come from [58].

Figure 16 presents the change in the utility of both ITDM and PDM over 29 negotiation rounds, and Figure 17 shows the proposed power profiles from each module during negotiation. The distance between the profiles proposed by ITDM and PDM is showed in Figure 18. The distance is defined as the inversion of Pearson correlation. With T being the time window, the distance between two profiles $x = \{x_1, \ldots, x_T\}$
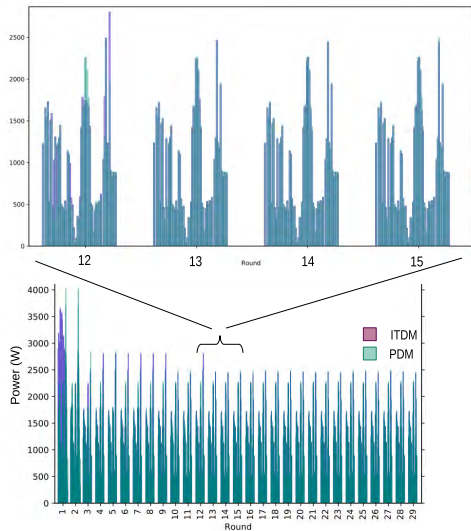
---

[5]https://www.openstack.org/

FIGURE 17. Best matching profiles proposed by ITDM and PDM for each round of negotiation.
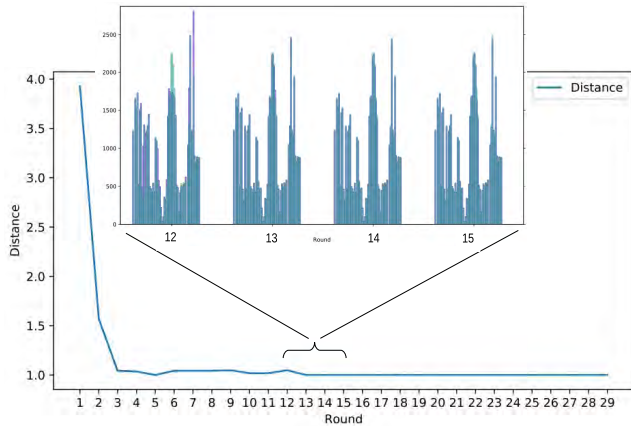

FIGURE 18. Distance between the best matching ITDM and PDM profiles over the negotiation rounds.

and $y = \{y_1, \ldots y_T\}$ is given as

$$\frac{\sqrt{\sum_{i=1}^{T}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{T}(y_i - \bar{y})^2}}{\sum_{i=1}^{T}(x_i - \bar{x})(y_i - \bar{y})},$$

where $\bar{x}$ and $\bar{y}$ are the average power levels of x and y, respectively.

In the figures, at the early negotiation rounds, the utilities of ITDM and PDM fluctuate, and the distance between their profiles is large. After negotiating by exchanging hints, the ITDM and PDM gradually generate more similar profiles, a compromise for both; then the utilities head toward stable points, and the distance is reduced. As in Figure 17, starting at round 13, the ITDM profile and PDM profile are relatively close to each other; their corresponding distance in Figure 18 also reaches a lower value and stops changing. In our designed algorithm, the negotiation is stopped when either a maximum number of iterations (i.e., 29) is reached,

TABLE 2. Performance comparison.

Scenario	Avg. exe. time (min)	Avg. prof. dist.	Avg. util. (ITDM util. in Euro, PDM util. in Euro)	Tot. util.	Avg. over. SLA violat. (%)
Best-Fit	3.16	-	(144.30, -101.66)	42.64	2.89
GreenSlot	0.43	-	(67.67, 46.30)	113.97	2.95
Negotiation $\varepsilon = 1.2$	10.69	1.17	(185.88, 125.92)	311.8	0.81
Negotiation $\varepsilon = 1.12$	16.31	1.11	(245.90, 118.77)	364.67	0.23

or the distance between ITDM profile and PDM profile is smaller than a threshold, or that distance remains unchanged for a predefined number of iterations. Note that, in the presented experiments, we have disabled the second and the third conditions in order to validate the convergence and stability of the negotiation algorithm.

The algorithm converges starting at round 13, then we can implement the PDM profile as the final solution. We implement the PDM profile because it is the feasible power that the PDM can supply. If there is a gap between the final ITDM and PDM profiles, the ITDM still has to accept the PDM power supply.

2) PERFORMANCE COMPARISON

We compare the results of our negotiation algorithm with the IT-leaded Best-Fit (Algorithm 1) described in detail in [5], and the well-known algorithm GreenSlot [19]. In the Best-Fit algorithm, the IT and electrical managements are not jointly considered. Instead, the IT workloads are scheduled while taking into account energy availability. With Greenslot, we implemented its "GreenVarPrices" version with several simplifications and modifications, making the configurations of machine and workload comparable with the two other algorithms. The modifications are: (i) the renewable energy availability is predicted without error, (ii) the task execution time is predefined, (iii) we still schedule a task after its due date, and (iv) a machine can execute multiple jobs at once. We ran the experiments on a local computer that has one Intel ® processor 2.20GHz with 4 cores, and 8.27GB memory. The workload includes 312 jobs of batch and service. We denote ε as the distance threshold between the ITDM profile and PDM profile. This threshold is used to adjust the stopping criterion of the negotiation algorithm.

In Table 2, we show the performances of the three algorithms on average execution time (avg. exe. time), average profile distance (avg. prof. dist.), average utility (avg. util.), total utility (tot. util.), and average overall SLA violation (avg. over. SLA violat.). The utility associated with each profile is the monetary benefit that this profile provides to

the ITDM or PDM. The total utility is the sum of ITDM and PDM utility; they are all in *Euro*. SLA violation is the percentage of violated jobs, i.e., due date violation for batch jobs and resource under-provisioning for service jobs. The table shows that the negotiation algorithm outperforms the two others in terms of utility and SLA violation. However, since the negotiation algorithm is an iterative approach, instead of a one-shot approach, it requires longer execution time. In addition, with a lower distance threshold, we obtain better results of profile distance, utility and SLA violation, but at the cost of a higher execution time. The PDM with the Best-Fit algorithm achieves negative utility because this method only considers IT scheduling. Note that the profile distances of the Best-Fit and Greenslot are not shown because these two methods produce only one profile.

VII. RELATED WORK

The high adoption in the usage of cloud datacenter by both private and public initiatives has increased significantly. In 2010, datacenters were consuming about 1.3% of world's electricity [36]. This high energy consumption results not only in large electricity costs but also incurs high carbon emissions [15]. To cope with this problem, several efforts have been conducted, mainly aiming to optimize the IT load placement, and to power datacenters partially with renewable energy sources. In this section, we present some of these initiatives separated in renewable-related approaches and projects (such as DATAZERO) that tackle this problem.

A. RENEWABLE ENERGY RELATED APPROACHES

Several works in the literature, such as the ones presented in the survey of Kong and Liu [35], aim to reduce the energy consumption of datacenters, or increase the renewable part. Here, we focus on the ones that are closely related to the DATAZERO project, and how they compare to it.

The research from Goiri et al., comprising GreenSlot [19], GreenHadoop [22], GreenSwitch [21] and Parasol [20], focuses on building research platforms for green datacenters. GreenSlot is a batch job scheduler for a datacenter powered by photo-voltaic panels and the electrical grid, whereas GreenHadoop is a framework that deals with MapReduce jobs. Both aim at maximizing the green energy consumption. Parasol is a prototype of a green datacenter with solar energy, batteries, and net metering. Additionally, the authors introduced GreenSwitch, a dynamic scheduler for workload and energy sources.

Some authors focus on maximizing the renewable energy usage in order to reduce the brown part, such as [43] which considers scheduling with resources over-commitment, [44] considering opportunistic usage of on-site renewable energy sources, [1] and [7] that consider shift of the workload according to the weather condition forecast, and [62] which migrate and consolidate the workload to keep the power discharge level at a level considered acceptable. In [23], Grange et al. propose a scheduling approach which considers the preferences of an abstract power infrastructure for a given planned power consumption. This preference indicator can reflect the usage of renewable energy, but the approach does not require prior knowledge of the way it is computed nor of the power infrastructure model.

Other authors such as [18], [32], [38], [39] and [52] focus on multi-objective approaches. The objectives vary, in some cases aiming at minimizing the energy consumption and the tasks violations [32], others at reducing the difference between available and consumed power, while maintaining the Quality of Service [52]. Some approaches [38], [39] also consider the grid electricity price in an approach that aims to minimize the cost and maximize the green energy consumption, or in the last case also aiming at minimizing the tasks makespan.

When compared to DATAZERO, the previously mentioned approaches are primarily focused on the IT side, with only a few information on the power side, such as the forecasted renewable production. At the exception of [23], these works aim to solve the problem in a centralized way, often without taking any decision for the electrical infrastructure which is considered as passive. Only [23] explicitly considers a separated management of the power infrastructure to avoid a central optimization, but without proposing power-side optimization methods and with an approach limited to batch tasks. While a few of them, such as [20], use a more detailed electrical model and take into consideration the battery charging state, the proposed centralized optimization method is unpractical or not always feasible for larger datacenters with thousands of machines and multiple energy storage elements. To tackle this problem, DATAZERO introduces a negotiation loop with a decentralized optimization of IT and power.

Several works aim at the design and implementation of more efficient datacenters. Some focused on matching renewable energy supply and demand/load in a certain time period [2], [42], evaluating traces of real datacenters and key metrics to make them more sustainable [41], or enabling carbon-constrained approaches with the aid of batteries and renewable energy [40]. DATAZERO proposes not only a more sustainable datacenter, but also a breakthrough infrastructure where all the energy comes from local renewable energy sources, while dealing with the challenges that it brings. Moreover, it proposes a running middleware able to mix simulation and real platforms for both IT and Power sides. This gives more possibilities to easily explore alternative scenarios, test alternative computing or power delivery components.

In more recent studies, some authors [5], [33] and [59] focus on datacenters powered only by renewable energy, with no connection to the grid. [33] minimizes makespan in HPC tasks, while [5] minimizes the number of due date violations for batch tasks, in both cases constrained by a power envelope. Sharma et al. [59] propose a more optimistic approach where web applications would not suffer from the regular on/off power cycles of the machines (called Blink by the authors), while still constrained only by renewable energy. In these terms, the DATAZERO approach introduces

a negotiation loop, not forcing the IT part only to adapt, but trying to find the best compromise between IT and power parts. By using different time horizons for the negotiation, it can also balance short-term decision with a long-term optimization goal. Furthermore, we consider application phases for a finer tuning of power demands over the time. We also consider application degradation in a more realistic manner and the impact that a degradation would cause for the user side.

Other existing approaches consider a "follow the renewable" concept [45], [46], by balancing the load among several datacenters and using the right mix of renewable energies. While DATAZERO considers only a single datacenter with local renewable sources, these are complementary approaches that may be used as a base to extend DATAZERO to multiple, geographically distributed datacenters.

B. EXISTING PROJECTS

In this subsection, we focus on projects in the same line as DATAZERO that aim at integrating renewable energy in datacenters. Projects such as GreenDataNet [24] provides a detailed study of necessary hardware, software and prediction models to efficiently integrate local PV resources into datacenters. From the IT perspective, the researchers use a correlation-aware VM placement (based on CPU load) which also aims at minimizing the energy consumption.

The DC4Cities [34] project considers a smart city with multiple datacenters connected to a smart grid. The idea is to reorganize the workload to match the shape of renewable energy available. The project also introduces two new metrics to evaluate effectiveness of software execution and renewable energy efficiency, a greedy algorithm that optimize task placement in DC under a power budget and an energy management system that acts as mediator between DC and the energy system.

RenewIT [57] presents a simulation tool to evaluate the energy performance of different technical solutions integrating renewable energy sources in several climate regions, reducing the carbon footprint of datacenters. The projects also contemplates the redistribution of VMs and consolidation of hosts using a constraint satisfaction library (Clopla) that is periodically called aiming to minimize energy consumption.

EPOC [3] focuses on energy-aware task execution from the hardware to the application's components, in the context of a mono-site and small datacenter connected to the regular electric grid and to local-renewable-energy sources (wind turbines or solar panels). The project considers VM migration and host consolidation in order to reduce energy consumption.

SeDuCe [14] (Sustainable Data Centers: Bring Sun, Wind and Cloud Back Together) aims to design an experimental infrastructure dedicated to the study of datacenters with low energy footprint. The idea of the project is to adapt the workload to the available amount of energy, coming from solar panels, wind turbines and batteries.

VIII. CONCLUSION

In this paper, we have introduced DATAZERO, a project aiming to provide consistent solutions for ensuring high availability of IT services and avoiding unnecessary redundancies, under the constraints of the intermittent nature of green power and services flows. The main originality of the project is to propose a negotiation module, between the IT and power control, that aims at finding a trade-off between the objectives and constraints of both parts instead of trying to solve a global optimization problem. To make this negotiation possible, we propose accurate electrical and IT models: we detailed the necessary need for electrical sources profiling, including advanced features like aging, Cloud application phases and degradation modes, IT server power and performance modeling and analysis. We then showed how optimizing such an infrastructure can be tackled both from the electrical and the IT point of view and how those two interact through the negotiation process. Last, we exhibited our proposal for a middleware based on ActiveMQ able to mix at once full simulations (using a Datacenter Workload simulator and a Power Simulator), experiments on real platforms (real power sources and IT servers running Openstack) and emulations (PHIL).

Future work consists in evaluating the different models and run experiments on the middleware platform, in order to assess the quality of the models and the robustness of the optimization algorithms coping with the uncertainty of power production and service demands. The negotiation algorithm itself must be tuned and several approaches, such as game theory, are assessed. Further, the introduction of uncertainty in the decision process or aging models for the components to take more reliable decisions will be two interesting research problems.

REFERENCES

[1] B. Aksanli, J. Venkatesh, L. Zhang, and T. Rosing, "Utilizing green energy prediction to schedule mixed batch and service jobs in data centers," in *Proc. 4th Workshop Power-Aware Comput. Syst. (HotPower)*, New York, NY, USA, 2011, pp. 5:1–5:5.

[2] M. Arlitt, C. Bash, S. Blagodurov, Y. Chen, T. Christian, D. Gmach, C. Hyser, N. Kumari, Z. Liu, M. Marwah, A. McReynolds, C. Patel, A. Shah, Z. Wang, and R. Zhou, "Towards the design and operation of net-zero energy data centers," in *Proc. 13th InterSoc. Conf. Therm. Thermomech. Phenomena Electron. Syst.*, May/Jun. 2012, pp. 552–561.

[3] N. Beldiceanu, B. D. Feris, P. Gravey, S. Hasan, C. Jard, T. Ledoux, Y. Li, D. Lime, G. Madi-Wamba, J.-M. Menaud, P. Morel, M. Morvan, M.-L. Moulinard, A.-C. Orgerie, J.-L. Pazat, O. Roux, and A. Sharaiha, "The EPOC project: Energy proportional and opportunistic computing system," in *Proc. Int. Conf. Smart Cities Green ICT Syst. (SMARTGREENS)*, May 2015, pp. 1–7.

[4] H. Casanova, A. Giersch, A. Legrand, M. Quinson, and F. Suter, "Versatile, scalable, and accurate simulation of distributed applications and platforms," *J. Parallel Distrib. Comput.*, vol. 74, no. 10, pp. 2899–2917, Oct. 2014.

[5] S. Caux, P. Renaud-Goud, G. Rostirolla, and P. Stolf, "IT optimization for datacenters under renewable power constraint," in *Proc. 24th Eur. Conf. Parallel Process.*, Turin, Italy, Aug. 2018, pp. 339–351.

[6] S. J. Chapin, W. Cirne, D. G. Feitelson, J. P. Jones, S. T. Leutenegger, U. Schwiegelshohn, W. Smith, and D. Talby, "Benchmarks and standards for the evaluation of parallel job schedulers," in *Proc. Workshop Job Scheduling Strategies Parallel Process.* Berlin, Germany: Springer, 1999, pp. 67–90.

[7] T. Cioara, I. Anghel, M. Antal, S. Crisan, and I. Salomie, "Data center optimization methodology to maximize the usage of locally produced renewable energy," in *Proc. Sustain. Internet ICT Sustainability (SustainIT)*, Apr. 2015, pp. 1–8.

[8] *Datazero D2.4 Deliverable: Sources and Material Profiling, V2*, DATAzero Consortium, Nea Smyrni, Greece, Dec. 2017.

[9] *Datazero D4.1 Deliverable: Model of the Optimization Problem of Data Centers. Cartography of the Optimization Tools*, DATAzero Consortium, Nea Smyrni, Greece, Apr. 2017.

[10] G. Cook, "How clean is your cloud?" Greenpeace Int., Amsterdam, The Netherlands, Tech. Rep., Apr. 2012. [Online]. Available: http://www.greenpeace.nl/Global/nederland/2012/publicaties/HowCleanisYourCloud.pdf

[11] G. Cook, T. Dowdall, D. Pomerantz, and Y. Wang, "Clicking clean: How companies are creating the green Internet," Greenpeace, Amsterdam, The Netherlands, Tech. Rep., Apr. 2014. [Online]. Available: http://media.oregonlive.com/silicon-forest/other/Clicking%20Clean%20Report.pdf

[12] G. Da Costa, L. Grange, and I. De Courchelle, "Modeling and generating large-scale Google-like workload," in *Proc. 7th Int. Green Sustain. Comput. Conf. (IGSC)*, Nov. 2016, pp. 1–7.

[13] G. Da Costa, J.-M. Pierson, and L. Fontoura-Cupertino, "Mastering system and power measures for servers in datacenter," *Sustain. Comput., Inform. Syst.*, vol. 15, pp. 28–38, Sep. 2017.

[14] *Seduce Project Press Announce*. Accessed: Jul. 30, 2019. [Online]. Available: https://seduce.fr/

[15] W. Deng, F. Liu, H. Jin, B. Li, and D. Li, "Harnessing renewable energy in cloud datacenters: Opportunities and challenges," *IEEE Netw.*, vol. 28, no. 1, pp. 48–55, Jan./Feb. 2014.

[16] M. E. M. Diouri, G. L. T. Chetsa, O. Glück, L. Lefèvre, J.-M. Pierson, P. Stolf, and G. Da Costa, "Energy efficiency in high-performance computing with and without knowledge of applications and services," *Int. J. High Perform. Comput. Appl.*, vol. 27, no. 3, pp. 232–243, Jul. 2013. doi: 10.1177/1094342013495304.

[17] F. Cupertino, G. Da Costa, A. Oleksiak, W. Piątek, J.-M. Pierson, J. Salom, L. Sisó, P. Stolf, H. Sun, and T. Zilio, "Energy-efficient, thermal-aware modeling and simulation of data centers: The CoolEmAll approach and evaluation results," *Ad Hoc Netw. J.*, vol. 25, pp. 535–553, Feb. 2015.

[18] M. Ghamkhari and H. Mohsenian-Rad, "Energy and performance management of green data centers: A profit maximization approach," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 1017–1025, Jun. 2013.

[19] Í. Goiri, R. Beauchea, K. Le, T. D. Nguyen, M. E. Haque, J. Guitart, J. Torres, and R. Bianchini, "GreenSlot: Scheduling energy consumption in green datacenters," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, Nov. 2011, pp. 1–11.

[20] Í. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, "Parasol and GreenSwitch: Managing datacenters powered by renewable energy," *ACM SIGARCH Comput. Archit. News*, vol. 41, no. 1, pp. 51–64, 2013.

[21] Í. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, "Designing and managing data centers powered by renewable energy," *IEEE Micro*, vol. 34, no. 3, pp. 8–16, May 2014.

[22] Í. Goiri, K. Le, T. D. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "GreenHadoop: Leveraging green energy in data-processing frameworks," in *Proc. 7th ACM Eur. Conf. Comput. Syst. (EuroSys)*, New York, NY, USA, 2012, pp. 57–70.

[23] L. Grange, G. Da Costa, and P. Stolf, "Green IT scheduling for data center powered with renewable energy," *Future Gener. Comput. Syst.*, vol. 86, pp. 99–120, Sep. 2018.

[24] *GreenDataNet Research Project*. Accessed: Jul. 30, 2019. [Online]. Available: http://www.greendatanet-project.eu

[25] *Clicking Clean: Who Is Winning the Race to Build a Green Internet?* Greenpeace, Amsterdam, Netherlands, Jan. 2017.

[26] T. Guérout, S. Medjiah, G. Da Costa, and T. Monteil, "Quality of service modeling for green scheduling in clouds," *Sustain. Comput., Inform. Syst.*, vol. 4, no. 4, pp. 225–240, Dec. 2014.

[27] M. Haddad, J.-M. Nicod, and M.-C. Péra, "Hydrogen infrastructure: Datacenter supply-refueling station synergy," in *Proc. IEEE Vehicle Power Propuls. Conf. (VPPC)*, Dec. 2017, pp. 1–6.

[28] M. Haddad, J.-M. Nicod, C. Varnier, and M.-C. Péra, "Optimisation de l'engagement des sources pour un Data Center approvisionné en énergie renouvelable," in *Proc. COMPAS*, Toulouse, France, 2017, pp. 1–8.

[29] H. Hlavacs, G. Da Costa, and J.-M. Pierson, "Energy consumption of residential and professional switches," in *Proc. Int. Conf. Comput. Sci. Eng. (CSE)*, Vancouver, BC, Canada, Aug. 2009, pp. 240–246. [Online]. Available: http://www.computer.org

[30] T. Horvath, T. Abdelzaher, K. Skadron, and X. Liu, "Dynamic voltage scaling in multitier Web servers with end-to-end delay control," *IEEE Trans. Comput.*, vol. 4, no. 4, pp. 444–458, Apr. 2007.

[31] Gurobi Optimization. (2016). *Gurobi Optimizer Reference Manual*. Accessed: Jul. 30, 2019. [Online]. Available: http://gurobi.com

[32] S. Iturriaga and S. Nesmachnow, "Scheduling energy efficient data centers using renewable energy," *Electronics*, vol. 5, no. 4, p. 71, 2016.

[33] A. Kassab, J.-M. Nicod, L. Philippe, and V. Rehn-Sonigo, "Scheduling independent tasks in parallel under power constraints," in *Proc. 46th Int. Conf. Parallel Process. (ICPP)*, Aug. 2017, pp. 543–552.

[34] S. Klingert, F. Niedermeier, C. Dupont, G. Giuliani, T. Schulze, and H. de Meer, "Renewable energy-aware data centre operations for smart cities the DC4Cities approach," in *Proc. Int. Conf. Smart Cities Green ICT Syst. (SMARTGREENS)*, May 2015, pp. 1–9.

[35] F. Kong and X. Liu, "A survey on green-energy-aware power management for datacenters," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 30:1–30:38, Jan. 2014.

[36] J. Koomey, "Growth in data center electricity use 2005 to 2010," *Rep. Anal. Press, Completed Request New York Times*, vol. 9, p. 161, Aug. 2011.

[37] K. Kurowski, A. Oleksiak, W. Piątek, T. Piontek, A. Przybyszewski, and J. Węglarz, "DCworms—A tool for simulation of energy efficiency in distributed computing infrastructures," *Simul. Model. Pract. Theory*, vol. 39, pp. 135–151, Dec. 2013.

[38] H. Lei, R. Wang, T. Zhang, Y. Liu, and Y. Zha, "A multi-objective co-evolutionary algorithm for energy-efficient scheduling on a green data center," *Comput. Oper. Res.*, vol. 75, pp. 103–117, Nov. 2016.

[39] H. Lei, T. Zhang, Y. Liu, Y. Zha, and X. Zhu, "SGEESS: Smart green energy-efficient scheduling strategy with dynamic electricity price for data center," *J. Syst. Softw.*, vol. 108, pp. 23–38, Oct. 2015.

[40] C. Li, Y. Hu, R. Zhou, M. Liu, L. Liu, J. Yuan, and T. Li, "Enabling datacenter servers to scale out economically and sustainably," in *Proc. 46th Annu. IEEE/ACM Int. Symp. Microarchitecture (MICRO)*, New York, NY, USA, 2013, pp. 322–333.

[41] C. Li, A. Qouneh, and T. Li, "Characterizing and analyzing renewable energy driven data centers," *SIGMETRICS Perform. Eval. Rev.*, vol. 39, no. 1, pp. 131–132, Jun. 2011.

[42] C. Li, A. Qouneh, and T. Li, "iSwitch: Coordinating and optimizing renewable energy powered server clusters," *SIGARCH Comput. Archit. News*, vol. 40, no. 3, pp. 512–523, Jun. 2012.

[43] Y. Li, A.-C. Orgerie, and J.-M. Menaud, "Opportunistic scheduling in clouds partially powered by green energy," in *Proc. IEEE Int. Conf. Data Sci. Data Intensive Syst.*, Dec. 2015, pp. 448–455.

[44] Y. Li, A.-C. Orgerie, and J.-M. Menaud, "Balancing the use of batteries and opportunistic scheduling policies for maximizing renewable energy consumption in a cloud data center," in *Proc. 25th Eur. Int. Conf. Parallel, Distrib. Netw.-Based Process. (PDP)*, Mar. 2017, pp. 408–415.

[45] M. Lin, Z. Liu, A. Wierman, and L. L. H. Andrew, "Online algorithms for geographical load balancing," in *Proc. Int. Green Comput. Conf. (IGCC)*, Jun. 2012, pp. 1–10.

[46] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. H. Andrew, "Geographical load balancing with renewables," *SIGMETRICS Perform. Eval. Rev.*, vol. 39, no. 3, pp. 62–66, Dec. 2011.

[47] U. Lublin and D. G. Feitelson, "The workload on parallel supercomputers: Modeling the characteristics of rigid jobs," *J. Parallel Distrib. Comput.*, vol. 63, no. 11, pp. 1105–1122, 2003.

[48] M. Marinelli, P. Maule, A. N. Hahmann, O. Gehrke, P. B. Nørgrd, and N. A. Cutululis, "Wind and photovoltaic large-scale regional models for hourly production evaluation," *IEEE Trans. Sustain. Energy*, vol. 6, no. 3, pp. 916–923, Jul. 2015.

[49] T. Mastelic, A. Oleksiak, H. Claussen, I. Brandic, J.-M. Pierson, and A. V. Vasilakos, "Cloud computing: Survey on energy efficiency," *ACM Comput. Surv.*, vol. 47, no. 2, Jan. 2015, Art. no. 33.

[50] R. B. Myerson, *Game Theory*. Cambridge, MA, USA: Harvard Univ. Press, Mar. 2013.

[51] A. Neffati, M. Guemri, S. Caux, and M. Fadel, "Energy management strategies for multi source systems," *Electr. Power Syst. Res.*, vol. 102, pp. 42–49, Sep. 2013. doi: 10.1016/j.epsr.2013.03.008.

[52] S. Nesmachnow, C. Perfumo, and Í. Goiri, "Holistic multiobjective planning of datacenters powered by renewable energy," *Cluster Comput.*, vol. 18, no. 4, pp. 1379–1397, Dec. 2015.

[53] *Openstack Cloud Platform*. Accessed: Jul. 30, 2019. [Online]. Available: https://www.openstack.org/

[54] A. C. Orgerie, M. Dias de Assunção, and L. Lefèvre, "A survey on techniques for improving the energy efficiency of large-scale distributed systems," *ACM Comput. Surv.*, vol. 46, no. 4, Apr. 2014, Art. no. 47.

[55] D. O'Sullivan, "Data centre redundancy levels," EATON, Dublin, Ireland, Tech. Rep., Mar. 2016.

[56] *Report to Congress on Server and Data Center Energy Efficiency*, U.S. Environ. Protection Agency ENERGY STAR Program, EPA, Washington, DC, USA, Tech. Rep., Aug. 2007.

[57] J. Salom, T. Urbaneck, E. Oró, *Advanced Concepts for Renewable Energy Supply of Data Centres* (River Publishers Series in Renewable Energy).

[58] M. Sengupta, Y. Xie, A. Lopez, A. Habte, G. Maclaurin, and J. Shelby, "The national solar radiation data base (NSRDB)," *Renew. Sustain. Energy Rev.*, vol. 89, pp. 51–60, Jun. 2018.

[59] N. Sharma, S. Barker, D. Irwin, and P. Shenoy, "Blink: Managing server clusters on intermittent power," *SIGARCH Comput. Archit. News*, vol. 39, no. 1, pp. 185–198, Mar. 2011.

[60] H. Sun, P. Stolf, and J.-M. Pierson, "Spatio-temporal thermal-aware scheduling for homogeneous high-performance computing datacenters," *Future Gener. Comput. Syst.*, vol. 71, pp. 157–170, Jun. 2017.

[61] M. vor dem Berge, W. Christmann, E. Volk, S. Wesner, A. Oleksiak, T. Piontek, G. Da Costa, and J.-M. Pierson, "CoolEmAll—Models and tools for optimization of data center energy-efficiency," in *Proc. Sustain. Internet ICT Sustainability (SustainIT)*, Oct. 2012, pp. 1–5.

[62] X. Wang, Z. Du, Y. Chen, and M. Yang, "A green-aware virtual machine migration strategy for sustainable datacenter powered by renewable energy," *Simul. Model. Pract. Theory*, vol. 58, pp. 3–14, Nov. 2015.

[63] C.-M. Wu, R.-S. Chang, and H.-Y. Chan, "A green energy-efficient scheduling algorithm using the DVFS technique for cloud datacenters," *Future Gener. Comput. Syst.*, vol. 37, pp. 141–147, Jul. 2014.

GWILHERM BAUDIC received the master's degree from ISAE-ENSICA, Toulouse, France, in 2013, and the Ph.D. degree from INPT, Toulouse, in 2016. He is currently a Research Engineer with the IRIT Laboratory. He is currently working on the ANR DATAZERO project, studying how to manage the electricity and IT services in renewable powered datacenters and the ANR e-Flooding project aimed to model fast floods in terms of risk management and impact on the infrastructures.

STÉPHANE CAUX was born in 1970. He received the Ph.D. degree in robotics from the University of Montpellier II, France, in 1997. Since 1998, he has been an Assistant Professor with the Université de Toulouse, France, where he is currently a Professor with the Laboratoire Plasma et Conversion d'Energie, LAPLACE UMR 5213 CNRS/INPT/UPS. His main research interests include robust control and observers for electrical systems (electrical motors, inverters, and fuel cell systems), and real-time energy management for multisource electrical systems (optimization method and fuzzy management).

BERK CELIK (GS'14–M'17) received the B.Sc. and M.Sc. degrees from Ege University, Turkey, in 2011 and 2013, respectively, and the Ph.D. degree from the FEMTO-ST Institute, Université de Technologie de Belfort-Montbéliard, France. He is currently a Postdoctoral Researcher with LAPLACE, Université de Toulouse, France, in 2017. His current research interests include power systems, integration of renewable energies, demand response, and the coordination of smart homes.

JEAN-MARC PIERSON received the Ph.D. degree from ENS-Lyon, France, in 1996. He was an Associate Professor with the University Littoral Cote-d'Opale, Calais, from 1997 to 2001, then with INSA-Lyon, from 2001 to 2006. He has been a Full Professor in computer science with the University of Toulouse, France, since 2006. He is a member of the IRIT Laboratory and the Chair of the SEPIA Team on distributed systems. His main interest includes large-scale distributed systems. He serves on several PCs and editorial boards in the cloud, grid, pervasive, and energy-aware computing area. Since the last years, his researches focus on energy-aware distributed systems; in particular, monitoring, job placement and scheduling, virtual machines techniques, green networking, autonomic computing, and mathematical modeling. He was chairing the EU funded COST IC804 Action on "Energy Efficiency in Large Scale Distributed Systems" and participates in several national and European projects on energy efficiency in large scale distributed systems. He is currently chairing the ANR DATAZERO Project.

GEORGES DA COSTA received the Ph.D. degree from LIG, Grenoble, France, in 2005, and the Habilitation from University Paul Sabatier, Toulouse, France, in 2015. He is currently an Assistant Professor in computer science with the University of Toulouse. He is a member of the IRIT Laboratory. His research currently focus on energy-aware distributed systems. His research highlights are HPC and cloud computing, large scale energy-aware distributed systems, performance evaluation, and ambient systems. He serves on several PCs in the energy-aware systems, grid, and peer-to-peer fields. He was the Chair of the COST1305 working group on "Programming models and runtimes." He is currently working in the French funded project, Energumen, on scheduling moldable application on HPC systems and in the French funded project, Datazero, on scheduling in renewable powered datacenters.

LÉO GRANGE received the master's degree in computer science, specialized in distributed systems and safety critical software, from the University of Toulouse, where he is currently pursuing the Ph.D. degree in computer science with the Institut de Recherche en Informatique de Toulouse (IRIT). For his Ph.D. studies, he is working on energy-efficient scheduling for cloud computing, and more specifically for datacenters powered with on-site renewable energy sources.

MARWA HADDAD received the degrees in engineering in instrumentation and industrial maintenance from the National Institute of Applied Science and Technology (INSAT), Tunisia. She is currently pursuing the Ph.D. degree in computer science with the Automatic Control and Micro-Mechatronic Systems Department (AS2M), FEMTO-ST Institut, University of Franche-Comté. For her Ph.D. studies, she is working on the sizing and management optimization of a stand-alone hybrid renewable energy system to specifically power datacenters.

JÉRÔME LECUIVRE received the master's degree in engineering from the Ecole Nationale Supérieure d'Electricité et Mécanique Nancy (ENSEM), and the Ph.D. degree in computer science from Nancy University, France, in 1996. He leads for EATON, an engineering team that develops software suite integrating power devices, namely UPSs, ePDUs, and so on, in the IT infrastructure management (servers, storage, virtualization, cloud orchestrator, and so on).

JEAN-MARC NICOD received the Ph.D. degree from the Ecole Normale Supérieure of Lyon (LIP at ENS), in 1997, and the HDR from the Université of Franche-Comté (LIFC), in 2006. He is currently a Full Professor in computer science with the Automatic Control and Micro-Mechatronic Systems Department (AS2M), Institut FEMTO-ST, National Engineering Institute in Mechanics and Microtechnologies (ENSMM), Besançon, France. His research interests include parallel and distributed computing, distributed systems, algorithms, scheduling, and decision making for distributed heterogeneous systems.

LAURENT PHILIPPE received the Ph.D. degree from the University of Franche-Comté, in 1993. From 1993 to 2001, he was an Associate Professor with the University of Franche-Comté, where he has been a Full Professor of computer science. He is leading a research team on distributed computing at FEMTO-ST Institut. He heads the regional computing center. His main research interests include distributed systems, parallelism, and scheduling.

VERONIKA REHN-SONIGO received the Ph.D. degree from the Ecole Normale Supérieure, Lyon, in 2009, and the University of Passau. From 2009 to 2010, she was a Research Associate with the Computer Science Laboratory of Grenoble (LIG). She currently holds a position as an Associate Professor with the University of Franche-Comté. Her research interests include parallel algorithms, replica placement, renewable energy, as well as scheduling for distributed heterogeneous platforms.

ROBIN ROCHE (SM'18) received the Ph.D. degree in electrical engineering from the Université de Technologie de Belfort-Montbéliard (UTBM), Belfort, France, in 2012. He is currently an Associate Professor with the Université Bourgogne Franche-Comté and UTBM, with the FEMTO-ST Institute, and also with the FCLAB Research Center. He is the Chair of the Microgrids Group, FEMTO-ST. His research interests include microgrids, energy management strategies, as well as renewable energy and hydrogen energy storage integration.

GUSTAVO ROSTIROLLA received the bachelor's degree in computer engineering from Univates, Brazil, and the master's degree in applied computing from the University of Vale do 2Rio dos Sinos, Brazil. He is currently pursuing the Ph.D. degree with IRIT/LAPLACE, Toulouse, France, supported by the ANR DATAZERO project. His research interests include energy consumption, renewable energy sources, cloud computing, and smart cities.

AMAL SAYAH received the Ph.D. degree from LSI, Toulouse, France, in 1982. He is currently an Assistant Professor in computer science with the University of Toulouse. He is a member of the IRIT Laboratory. His research currently focus on energy-aware distributed systems and on autonomic systems that manage fast floods crisis. He has been involved in different research projects, such as the ACTION COST IC0804 "Energy Efficiency in Large Scale Distributed Systems," and the European FP7 CoolEmAll project. He is currently working on the ANR DATAZERO project, studying how to manage the electricity and IT services in renewable powered datacenters and the ANR e-Flooding project aimed to model fast floods in terms of risk management and impact on the infrastructures.

PATRICIA STOLF received the Ph.D. degree from INSA, Toulouse, France, in 2004. She is currently an Associate Professor in computer science with the University of Toulouse. She is a member of the IRIT Laboratory. Her main research interests include large scale distributed systems like grid or clouds, distributed algorithms, and autonomic computing. Her research currently focuses on resources management, load-balancing, and energy-aware and thermal-aware placement. She has been involved in different research projects, such as the ACTION COST IC0804 "Energy Efficiency in Large Scale Distributed Systems," the European CoolEmAll project, and the national ANR SOP project. She is currently working on the ANR DATAZERO project, and studying how to manage the electricity and IT services in a datacenter operated with several green energy sources.

CHRISTOPHE VARNIER is currently an Associate Professor with the Ecole Nationale Supérieure de Mécanique et des Microtechniques, Besançon, where he teaches computer science. He is a member of the PHM team of the Automatic Control and Micro-Mechatronic Systems (AS2M) Department, FEMTO-ST Institute. He works on prognostics and health management and more specifically on scheduling, planning, and optimization.

• • •

MINH-THUYEN THI received the M.E. and Ph.D. degrees in information and communications from Inje University, South Korea, in 2014 and 2017, respectively. He is currently a Postdoctoral Researcher with the University of Toulouse, France, working on distributed computing model for ANR DATAZERO project. His research interests include communication systems, wireless systems, distributed systems, and game theory.