

# An Optimization Model to Reduce Energy Consumption in Software-Defined Data Centers

Claudia Canali<sup>1</sup>, Riccardo Lancellotti<sup>1</sup>, and Mohammad Shojafar<sup>2</sup>

<sup>1</sup> Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, Modena, Italy

{claudia.canali, riccardo.lancellotti}@unimore.it

<sup>2</sup> Italian National Consortium for Telecommunications (CNIT), Rome, Italy  
mshojafar@cnit.it

**Abstract.** The increasing popularity of Software-Defined Network technologies is shaping the characteristics of present and future data centers. This trend, leading to the advent of Software-Defined Data Centers, will have a major impact on the solutions to address the issue of reducing energy consumption in cloud systems. As we move towards a scenario where network is more flexible and supports virtualization and softwarization of its functions, energy management must take into account not just computation requirements but also network related effects, and must explicitly consider migrations throughout the infrastructure of Virtual Elements (VEs), that can be both Virtual Machines and Virtual Routers. Failing to do so is likely to result in a sub-optimal energy management in current cloud data centers, that will be even more evident in future SDDCs. In this chapter, we propose a joint computation-plus-communication model for VEs allocation that minimizes energy consumption in a cloud data center. The model contains a threefold contribution. First, we consider the data exchanged between VEs and we capture the different connections within the data center network. Second, we model the energy consumption due to VEs migrations considering both data transfer and computational overhead. Third, we propose a VEs allocation process that does not need to introduce and tune weight parameters to combine the two (often conflicting) goals of minimizing the number of powered-on servers and of avoiding too many VE migrations. A case study is presented to validate our proposal. We apply our model considering both computation and communication energy contributions even in the migration process, and we demonstrate that that our proposal outperforms the existing alternatives for VEs allocation in terms of energy reduction.

**Keywords:** Cloud computing, Software-Defined Networks, Software-Defined Data Center, Energy consumption, Optimization model

## 1 Introduction

The fast deployment of new services involving more and more intense data traffic exchange and complex QoS demands require the underlying infrastructure of a cloud data center to be highly flexible and adaptable. This scenario motivates the paradigm shift

towards the adoption of techniques such as Software-Defined Network applied to data centers, giving origin to Software-Defined Data Centers (SDDCs), where virtualization is extended to network elements and functionalities. This change will have a major impact on the solutions to address the issue of reducing energy consumption in cloud systems, since SDDCs realize a more seamless integration of the network within the data center IT processes, opening up to the possibility of novel energy-efficient resource strategies for the cloud infrastructures integrating complex and adaptive network management.

Most of the existing solutions to reduce energy consumption in cloud data centers mainly focus on server consolidation, which aims at minimizing the number of turned on physical servers while satisfying the resource demands of the active Virtual Machines (VMs) [1–4]. However, an effective management of SDDCs must be network-aware, meaning the need to consider the impact of data traffic exchange between the Virtual Elements (VEs) of the cloud infrastructure, that may be both Virtual Machines and Virtual Routers. Failing to do so is likely to result in a sub-optimal energy management, because networks in modern data centers tend to consume about 10%-20% of energy in normal usage, and may account for up to 50% energy during low loads [5]. Furthermore, few studies proposing solutions for VEs allocation consider the contribution of VEs migration to energy consumption, both in terms of computational and network costs. On the other hand, when taking into account the costs for migration, this is carried out in a quite straightforward (e.g., [6]): the allocation model considers just the number of VEs migrations, and introduces weight parameters to address the trade-off of minimizing the number of turned on physical servers while reducing expensive VMs migrations required for the server consolidation. These limitations are clearly visible in modern data centers, but will be even more critical in future SDDCs, where the support for more flexible network reconfiguration allows the migration of both virtual machine and communication channels and virtualized network apparatus [7]. Hence, we believe that traditional allocation policies, that are network blind or adopt simplified models for migration, will be inadequate to support future SDDC infrastructures.

This chapter, which extends a previous study of the authors [8], presents a joint computation-plus-communication model for Virtual Elements (VEs) allocation that minimizes energy consumption in a SDDC scenario. The proposed optimization model for VMs allocation aims not only to reduce as much as possible the number of turned on servers, but also to minimize the energy consumption due to the exchange of data traffic between VMs over the data center infrastructure. The main contribution of our proposal is threefold. First, the model is aware of the heterogeneous connections existing within the data center and considers the data traffic exchanges occurring between VEs. Second, the energy contribution of VEs migration takes into account both the data transfer and the computational overhead. Third, the objective function of the proposed optimization model does not need to add weight parameters to merge the often conflicting goals of avoiding a high number of VEs migrations while guaranteeing the optimal VEs allocation.

We evaluate the performance of the proposed model by considering a case study based on traces from a real data center. The results confirm the validity of our model, showing that the proposed solution outperforms approaches for VEs allocation in terms

of energy reduction. Moreover, we show that our optimization model allows to limit the number of VM migrations, thus achieving more stable energy consumption over time and leading to major global energy saving if compared with other existing approaches.

The remainder of this paper is organized as follows. Section 2 presents the reference scenario for our proposal, while Section 3 describes the main requirements for an energy model in SDDCs. Section 4 describes the proposed model for solving the VEs allocation problem. Section 5 describes the case study based on traces from a real data center. Finally, Section 6 concludes the paper with some final remarks and outlines open research problems.

## 2 Reference scenario

In this section we describe the reference scenario for our proposal, comparing the characteristics of a Software-Defined Data Center (SDDC) with a more traditional cloud data center. Starting from this scenario, we illustrate how the characteristics of a SDDC impact on the model used to ensure the energy-efficient management of a cloud data center, with special focus on the operations that decide the allocation of the VMs over the physical servers of the infrastructure.

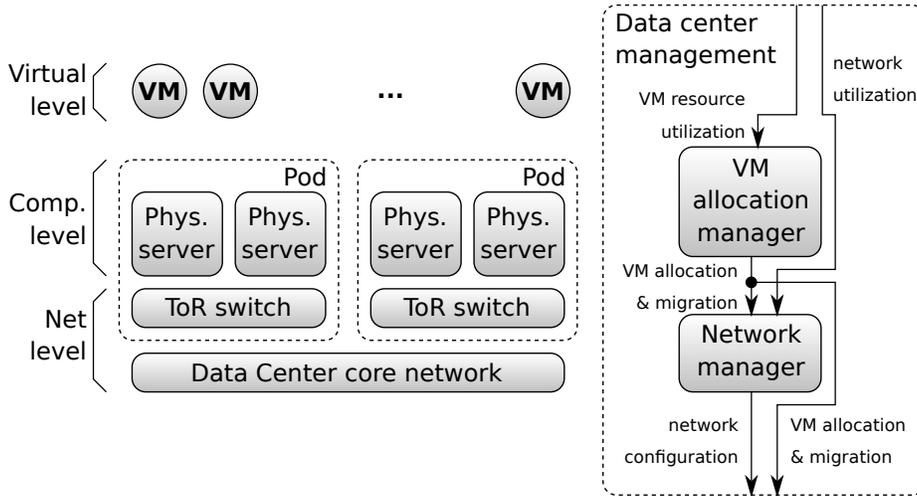


Fig. 1: Traditional Cloud Data Center

Figure 1 presents the general schema of a traditional cloud data center.

The considered data center is based on the Infrastructure as a Service (IaaS) paradigm, where VMs can be deployed and destroyed at any time by cloud customers. This explains the system view at a virtual level, where we simply have the VMs that are the object of the cloud customer attention. The cloud provider must map these VMs over the physical infrastructure, defining the data center management strategies. The

VMs are hosted on physical servers, which are grouped into pods (roughly corresponding in a set of interconnected racks housing the servers). The data center is based on a two-level network architecture, with *Top-of-Rack (ToR) switches* connecting the servers of the same pod, and an upper layer of networking (*data center core network*) that manages the communication among multiple racks of servers. This structure implies two different costs for transferring data between servers belonging to the same pod (passing through the ToR switch) and to different pods (passing through the data center core network).

The figure also presents the logical blocks of the system implementing the data center management logic (in the right part of Fig. 1). The input of such task is the data about the resource utilization of the VMs and the information about the network traffic over the data center. Two main components are involved in the management of the data center. The *VM allocation manager* (on the top-right side of Fig. 1) is the data center component responsible for running the model for VMs allocation that determines the optimal allocation of VMs on the physical servers to minimize the global energy consumption. After achieving a solution, the allocation manager notifies the servers of the VMs migrations that need to be applied. The VM allocation and migration decisions are then used as the input for a second block: the *Network manager* that, considering the current network utilization level, issues commands for re-configuring routing tables, firewalling rules and to turn on or off some network links to cope with the network traffic patterns changes resulting from the VM allocation manger decisions.

The VMs allocation manager operates mainly by planning VMs migrations across the infrastructure to accomplish the goal of minimizing the energy consumption of the cloud data center in terms of both computational and network contributions. It is worth to note that many solutions for energy-aware VMs allocation are based on reactive approaches, which rely on events to trigger the VM migrations [1, 6, 4]. On the other hand, in a SDDC scenario an approach based on time intervals, where a control of the optimality of the VMs allocation on the physical servers is periodically performed, could be preferable. The main reason is that, while for CPU utilization it is feasible and easy to define events (typically based on thresholds) to trigger migrations, for network-related energy costs it is much more difficult to define similar triggers. The details of the optimization model proposed to reduce energy consumption in the networked cloud data center are described in Section 4.

Figure 2 presents an example of a newer generation of infrastructure, called Software-Defined Data Center (SDDC), which is a data center where virtualization is much more pervasive within the system. A first significant difference with a traditional data center is evident if we compare the virtual level of Figures 1 and 2: while a traditional data center considers just VMs, in a SDDC we focus on virtual networks composed of *Virtual Elements (VE)* that can be either Virtual Machines (VMs) or Virtual Routers (VRs). A virtual router can be both implemented as a VM running a routing software, or as a set of rules and actions for a SDN appliance. This paradigm shift explains why there is a partial overlap between the computational and network layers in Fig. 2. The need to take into account the interaction between virtual elements in a virtual network and the inherent programmability of the network layer have a major impact on the data center management (right part of Fig. 2): in a SDDC the management is an

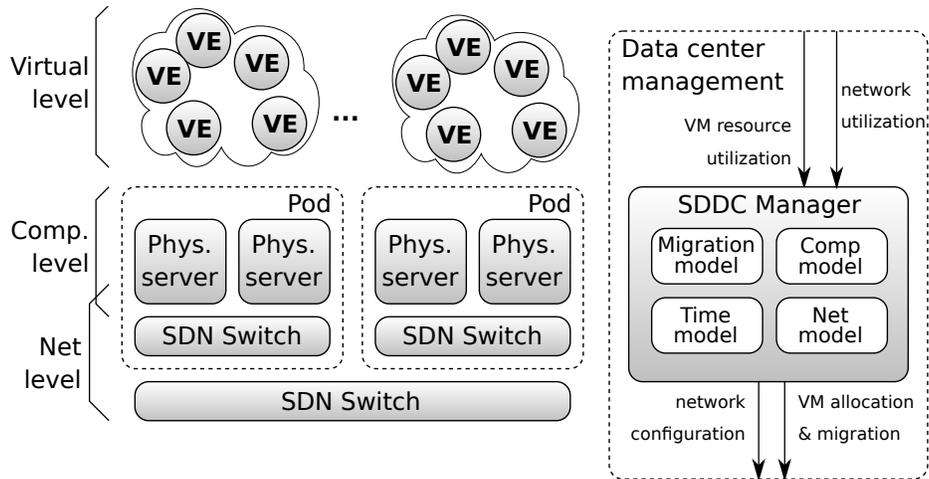


Fig. 2: Software-Defined Data Center

integrated process that combines different models for the multiple elements of the data center. In particular, we evidence four models that must be taken into account for the SDDC management: a *computational model*, a *network model*, a *migration model* for VMs and VRs, and a *time model*. These models (that we will discuss in more detail in the following section) take as input the data on computation and networking usage from the data center and produce an integrated decision about VMs/VRs allocation and migrations, and re-configuration of the network policies.

It is worth to note that considering the data center management as a single process is important to cope with the double nature of VRs and is the only viable choice to support the vision of virtual networks at the level of the virtual layer that characterize this scenario. It is also worth to note that a model that fully leverages the potential of SDN and router virtualization simplifies the adoption of common standards, such as the OpenFlow protocol<sup>3</sup>, that can be used both in physical and virtual SDN devices, and provides a common method to define heterogeneous network functionalities, ranging from network monitoring, to IP routing, and up to NAT and firewalling. Furthermore, the presence of virtualization simplifies both the management of migration (supporting traffic engineering techniques [9] to make sure that live VEs migration does not interfere with network operations) configuration savings and device status checkpointing operations (for example using VM snapshots), thus making faster and less error prone the recovery from a faulty state of the networking infrastructure.

### 3 Requirements for a SDDC energy model

The approaches for efficient energy management in cloud data centers can be classified according to how they model the four main components already underlined in the pre-

<sup>3</sup> <http://archive.openflow.org/wp/learnmore/>

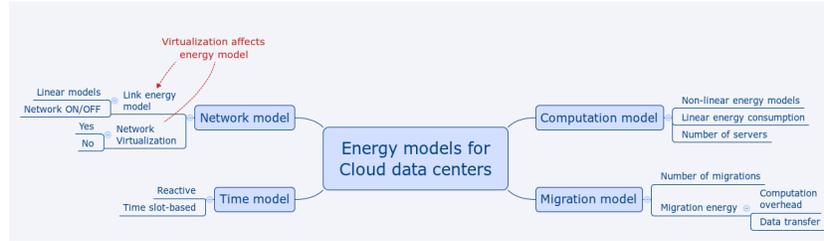


Fig. 3: Models for Energy Contributions in SDDC

vious section: (1) computation, (2) network, (3) VEs migration, and (4) time, as shown in Fig. 3 (and anticipated in the internal representation of the SDDC management in Fig. 1).

Existing solutions typically adopt simplified models not considering all the above components. Most of the studies just consider the energy consumption related to computational processes, not taking into account networking aspects in their proposed model [10, 11]. In this case, the main mechanism for optimizing energy consumption in the cloud data center is to operate a VEs allocation that minimizes the number of active physical hosts through server consolidation techniques [12, 10, 13]. However, this vision does not capture the challenges of SDDC with the increasing amount of data-intensive applications. Few studies consider the energy consumption related to the data transfer among VEs, and they typically focus only on optimizing the network infrastructure in terms of links reconfiguration or reallocating VEs based only on their traffic exchanges [14]. The rest of this section details the possible options in modeling the different contributions to energy consumption in a SDDC.

**Computation model.** Three main approaches are typically adopted to model computation-related energy. A possibility is to focus on a detailed model of CPU utilization that includes frequency scaling features (e.g., DVFS). These models typically introduce a *non-linear dependency* (e.g., quadratic, cubic or exponential) of energy consumption on the CPU frequency [15, 12]. However, this approach would lead to very complex models when a high level of virtualization is applied to computational tasks (through VMs or containers) as in large size cloud data centers. Moreover, it does not capture the fact that power consumption of several sub-systems, such as disks and memory, is not related to the actual CPU frequency. Hence, the most widespread option is to model the energy consumption at the level of physical host, considering the energy consumption starting from a minimum (idle) value and increasing with a *linear dependency* on the host utilization [1, 11]. The final option is the simplest approach, that just considers the *number of physical hosts* [13]. However, this latter model is hardly applicable in a SDDC, because of the inherent complexity of the data center (possibly composed of heterogeneous physical hosts) and of the difficulty of combining possible energy contributions expressed through different units of measurements (e.g., number of hosts and energy due to data transfers).

**Network model.** When network-related energy consumption is taken into account, multiple choices may affect the resulting model. The energy consumption due to traf-

fic exchange on data links may be modeled following two main approaches. The first approach considers the energy consumption as mainly related to the *status ON or OFF* of the network link [16]: this is appropriate when the link has a fixed data rate or when the main energy expenditure is due to features, such as Power-over-Ethernet, that are not affected by the volume of data exchanged over the link. Recent proposals suggest a second more realistic approach where energy is modeled as the sum of a fixed value, related to the status of the network interfaces, plus a variable contribution that depends on the actual link activity. This model has been proposed both to model the impact of energy consumption of data center network appliances [17] and to model the actual consumption of network link interfaces in the presence of *Adaptive Link Rate (ALR)* systems [14]. Another important consideration regards should be done about the presence of *Network Virtualization*. In this case, all or part of networking is implemented through virtual devices: for example, exploiting the added flexibility of Software-Defined Network elements [7], or running multiple virtual routers on a single physical router with virtualization capabilities [14], or using VMs to run the IP routing software [18]. In this case, the energy consumption of networking elements virtualized over highly programmable or general purpose hardware should be considered linearly proportional to the network utilization [19, 15] (similarly to virtualized computation over a general-purpose hardware).

**Migration model.** A migration process occurs when a virtual element (a VM or a VR) is moved from one host to another one. We should consider that migration is not a cost-free operation from an energy point of view: developing an energy model able to reduce the amount of migrations is a critical goal to achieve. Several optimization models simply consider the *number of VM migrations* in the objective function to minimize the data center energy consumption [12, 6]. However, this approach presents two main limitations: first, the assumption of identical virtual elements from a migration energy point of view; second, the need to use weight parameters to include the number of migrations in the energy budget of the data center. More sophisticated approaches have been recently proposed that consider two possible energy contributions of the migration process. The first contribution is the energy associated with the *data transfer* of the VE memory from one host to the another, possibly including also the management of dirty memory pages during a live migration. The second contribution is related to the *computational overhead* caused by the virtual element migration at the level of hypervisor, that may reach up to 10% for the duration of the migration process [11].

**Time model.** As regards time modeling, we have two main approaches: *reactive* and *time slot-based* models. In the reactive model, any action aiming to reduce energy consumption is triggered by some events. This model is suitable to cope with critical situations, such as overload of the physical hosts, while keeping a minimum level of reconfiguration operations carried out only when strictly required [1, 12, 4, 11]. On the other hand, the time slot-based model carries out a continuous optimization effort, where reconfigurations of the data center are periodically evaluated [10]. This approach is useful in situations where sub-optimal configurations of the data center would not trigger any reconfiguration in a reactive model but, over long period of time, cause significant energy waste. A reactive model is the most common choice when the energy optimization considers mainly computational resources. However, optimization of net-

work traffic across the data center is more often associated to a time slot-based approach for the difficulty to define network-related thresholds that trigger events. This motivates the preference towards the time slot-based model when features of the SDDC are taken into account (e.g., use of virtual routers).

As a concluding remark, we observe that none of the previously cited energy models, considered by itself, is suitable to fully exploit the opportunities and to capture the inherent complexity of a SDDC. Indeed, to achieve this goal the energy model needs to include all the four previously aspects with an adequate level of accuracy. On the other hand, each of the above considered models includes some oversimplification that makes it not completely suitable for SDDC case. This motivates our proposal of an example model that is explicitly tailored to this type of scenario.

## 4 Problem model

This section describes the proposed model to address the problem of optimizing the VEs allocation on the physical servers of the data center with the goal to reduce the energy consumption. Specifically, we take into account the contributions of computational and networking tasks as well as of VEs migrations. The energy model used for the contribution of VEs migration is a qualifying and original point of the proposed solution.

### 4.1 Model overview

The model considers a set of servers  $\mathcal{M}$ , where each server  $i$  hosts multiple VEs. For each VE  $j \in \mathcal{N}$ , we consider requirements expressed in terms of CPU power  $c_j$ , memory  $m_j$  and network traffic, that is described by the data traffic passing between two VEs  $d_{j_1, j_2}$ . In this model, we made the assumption that a SLA is respected if and only if the VE actually has the specified CPU, memory, and network resources (same approach in [1], where the SLA is based only the CPU resource). This assumption for the SLA is consistent with the Infrastructure as a Service (IaaS) paradigm.

As for the time model adopted in our proposal, we assume that time is modeled through a discrete succession of intervals, where each interval has the same length  $\mathcal{T}$ . It is worth to note that, differently from our approach, many other studies rely on a reactive model based on events: for example, an event is triggered when the CPU utilization exceeds a certain threshold [1, 6, 4]). We assume that the VEs resource demands refer to a generic time interval  $t$ , and that the VEs allocation over the physical servers of the data center is known from the previous time interval  $t - 1$ . The knowledge of the previous VEs allocation is a necessary condition to exploit an approach based on dynamic programming. On the other hand, the knowledge of the future VEs requirements in terms of CPU, memory and network traffic may be predicted on the basis of recurring resource demand patterns, such as diurnal patterns that are typical of the network traffic exchange among VEs within a data centers [10].

For a new VE entering the system at time interval  $t$ , we consider the nominal values of the resource requirements in terms of CPU and memory, and we discard inter-VEs communication costs since we cannot assume to have information about it. As the new

VE enters, it is placed in the system using the Modified Best Fit Decreasing (MBFD) algorithm presented in [1].

The optimization model is now described. The list of the main decision variables, model parameters and internal variables used in the formal description is presented in Table 1 (the table has been used also in a previous work of the authors [8]).

Table 1: Notation.

Symbol	Meaning/Role
<b>Decision variables</b>	
$x_{i,j}(t)$	Allocation of VE $j$ on server $i$ at time $t$
$O_i(t)$	Status (ON or OFF) of server $i$
<b>Model parameters</b>	
$x_{i,j}(t-1)$	Allocation of VE $j$ on server $i$ at time $t-1$
$\mathcal{T}$	Duration of a time interval
$\mathcal{N}$	Set of existing VEs to deploy $ \mathcal{N}  = N$
$\mathcal{M}$	Set of on servers in the data center $ \mathcal{M}  = M$
$c_j(t)$	Computational demand of VE $j$ at time $t$
$d_{j_1,j_2}(t)$	Data transfer rate between VE $j_1$ and $j_2$ at time $t$
$m_j(t)$	Memory requirement demand of VE $j$ at time $t$
$c_i^m$	Maximum computational resources of server $i$
$d_i^m$	Maximum data rate manageable by server $i$
$\mathcal{E}_{d_{i_1,i_2}}$	Energy consumption for transferring 1 data unit from $i_1$ to $i_2$
$m_i^m$	Maximum memory of server $i$
$P_i^m$	Maximum power consumption of server $i$
$P_i^d$	Power consumption related to the ‘‘on’’ status of network connection of server $i$
$K_{C_i}$	Ratio between maximum and idle power consumption of server $i$
$K_{M_i}$	Computational overhead when server $i$ is involved in a migration
<b>Model variables</b>	
$i$	Index of a server
$j$	Index of a VE
$\mathcal{E}_{C_i}(t)$	Energy for server $i$ at time $t$
$\mathcal{E}_D(t)$	Energy for data transfer for server $i$ at time $t$
$\mathcal{E}_{M_j}(t)$	Energy for migration of VE $j$ time $t$
$g_{i,j}^-(t)$	1 if VE $j$ migrates <i>from</i> server $i$ time $t$
$g_{i,j}^+(t)$	1 if VE $j$ migrates <i>to</i> server $i$ at time $t$

We recall that in our model the VEs migration is not only necessary for server consolidation, but it also allows us to reduce the energy contribution related to the data transfer between couples of VEs. The VEs migrations are modeled by means of two

matrices, whose elements  $g_{i,j}^-(t)$  and  $g_{i,j}^+(t)$  represent the source and destination of a migration (with  $i$  being the server and  $j$  the VE), similarly to [6].

The decision variables of the problem are: an allocation binary matrix, whose elements  $x_{i,j}(t)$  determine the allocation of the VE  $j$  on the server  $i$ , and a binary vector, where the elements  $O_i(t)$  represents the status of the physical server  $i$ , that can be ON ( $O_i(t) = 1$ ) or OFF ( $O_i(t) = 0$ ). Finally, the allocation matrix at time  $t - 1$  represents the system status at the *end* of the interval  $t - 1$ : this means that the VEs that left and joined the system in the interval  $t - 1$  have been removed and added in the allocation matrix.

We now detail the optimization problem defining the VEs allocation for the time interval  $t$ .

## 4.2 Optimization model for VEs allocation

In this section we formally describe the considered optimization problem. The objective function (1.1) of the optimization model aims to minimize the main three contributions to the energy consumption: *Computational demand*, *Data transfer*, and *VE migration*. The optimization model is now discussed in details, analyzing each of the three components. We recall that the objective function and the model constraints comes from the previous work of the authors [8].

$$\min \sum_{i \in \mathcal{M}} \mathcal{E}_{C_i}(t) + \mathcal{E}_D(t) + \sum_{j \in \mathcal{N}} \mathcal{E}_{M_j}(t) \quad (1.1)$$

subject to:

$$\sum_{j \in \mathcal{N}} x_{i,j}(t) c_j(t) \leq c_i^m O_i(t) \quad \forall i \in \mathcal{M}, \quad (1.2)$$

$$\begin{aligned} & \sum_{j_1 \in \mathcal{N}} \sum_{j_2 \in \mathcal{N}} (x_{i,j_1}(t) + x_{i,j_2}(t) - 2x_{i,j_1}(t)x_{i,j_2}(t)) d_{j_1,j_2}(t) \leq \\ & \leq d_i^m O_i(t), \quad \forall i \in \mathcal{M}, \end{aligned} \quad (1.3)$$

$$\sum_{j \in \mathcal{N}} x_{i,j}(t) m_j(t) \leq m_i^m O_i(t), \quad \forall i \in \mathcal{M}, \quad (1.4)$$

$$\sum_{i \in \mathcal{M}} x_{i,j}(t) = 1, \quad \forall j \in \mathcal{N}, \quad (1.5)$$

$$\sum_{i \in \mathcal{M}} g_{i,j}^+(t) = \sum_{i \in \mathcal{M}} g_{i,j}^-(t) \leq 1, \quad \forall j \in \mathcal{N}, \quad (1.6)$$

$$g_{i,j}^-(t) \leq x_{i,j}(t-1), \quad \forall j \in \mathcal{N}, i \in \mathcal{M}, \quad (1.7)$$

$$g_{i,j}^+(t) \leq x_{i,j}(t), \quad \forall j \in \mathcal{N}, i \in \mathcal{M}, \quad (1.8)$$

$$x_{i,j}(t) = x_{i,j}(t-1) - g_{i,j}^-(t) + g_{i,j}^+(t), \quad \forall j \in \mathcal{N}, i \in \mathcal{M}, \quad (1.9)$$

$$x_{i,j}(t), g_{i,j}^+(t), g_{i,j}^-(t), O_i(t) = \{0, 1\}, \quad \forall j \in \mathcal{N}, i \in \mathcal{M}, \quad (1.10)$$

We model the *Computational demand* energy consumption of a generic server  $i$  as the sum of two components (as in [1]): first, a fixed energy cost for a server in the ON status ( $P_i^m K_{C_i}$  is the power consumption for an idle server); second, a variable cost

which is linearly proportional to the server utilization ( $P_i^m$  is the power consumption of a fully utilized server). The server utilization is computed based on the computational demands of each VE hosted on the server  $c_j(t)$  and the maximum server capacity ( $c_i^m$ ). The computational demand component is expressed as follows:

$$\mathcal{E}_{C_i}(t) = O_i(t) \mathcal{T} P_i^m \left( K_{C_i} + (1 - K_{C_i}) \frac{\sum_{j \in \mathcal{N}} x_{i,j}(t) c_j(t)}{c_i^m} \right)$$

As regards the *Data transfer* component, it represents a data center-wise value that we model again as the sum of two elements: the first component is represented by the power consumption of the server network interfaces when they are in an idle but turned on status ( $P_i^d$  for server  $i$ ); the second component is proportional to the amount of data exchanged and is based on the parameter  $d_{j_1, j_2}$  describing the data exchange between two VEs  $j_1$  and  $j_2$ ). This approach is consistent with the proposal presented in [19]. According to [1, 19, 10], we assume a linear energy model for the network data transfer: this model is realistic for current data centers and will be even more viable for modeling future data centers exploiting software-defined and virtualized network functions, where the network functions can be considered as abstract computation elements [7]. It is worth to note that the square matrix  $\mathcal{E}_{d_{i_1, i_2}}$ , representing the cost to transfer a data unit between two different servers, can capture the characteristics of any topology of a data center network. Hence, the global energy cost of the data exchange is described as follows:

$$\mathcal{E}_D(t) = \sum_{i \in \mathcal{M}} O_i(t) \mathcal{T} P_i^d + \sum_{j_1 \in \mathcal{N}} \sum_{j_2 \in \mathcal{N}} \sum_{i_1 \in \mathcal{M}} \sum_{i_2 \in \mathcal{M}} x_{i_1, j_1}(t) x_{i_2, j_2}(t) d_{j_1, j_2}(t) \mathcal{T} \mathcal{E}_{d_{i_1, i_2}}$$

The third component is the *VEs migration* cost, which represents a per-VE value modeling the energy consumption for the migration process. The energy contribution caused by the migration of a generic VE  $j$  depends on two main elements. First, the whole memory  $m_j$  of the migrating VE is transferred from a source to a destination server (note that the amount of data actually transferred is slightly higher than the nominal one because it is necessary to retransmit the dirty memory pages, but the typical small size of the active pages with respect to the VE global memory space allows us to neglect this effect). Second, we need to take into account the performance degradation occurring during the memory copy between two servers: we model the performance degradation through the parameter  $K_{M_{i,j}}$  for the server  $i$  hosting the VE  $j$ . The degradation is typically in the order of 10% of the performance [20] and takes typically a few tens of seconds, which is significantly lower with respect to the length of the time slot  $\mathcal{T}$ . The energy cost for the migration VE  $j$  can be described as:

$$\begin{aligned} \mathcal{E}_{M_j}(t) = & \sum_{i_1 \in \mathcal{M}} \sum_{i_2 \in \mathcal{M}} g_{i_1, j}^-(t) g_{i_2, j}^+(t) \left( m_j(t) \mathcal{E}_{d_{i_1, i_2}} + \right. \\ & \left. + (1 - K_{C_{i_1}}) P_{i_1}^m K_{M_{i_1}} \mathcal{T} + (1 - K_{C_{i_2}}) P_{i_2}^m K_{M_{i_2}} \mathcal{T} \right) \end{aligned}$$

It is important to note that the proposed model is significantly more complex with respect to state-of-the-art models considering only the number of VEs migrations, as done in [6]. The model complexity is motivated by the need to consider a complete

network model assuming that a VE migration should be triggered in the system only if the energy savings due to the better VE allocation in the future time slot will compensate the cost of the migration.

Let us now describe the constraints of our optimization problem. The capacity limit of the bin-packing problem of VEs allocation is expressed through the first group of constraints. The constraint 1.2 indicates that the sum of the CPU demands  $c_j(t)$  of the VEs allocated on each server must not exceed the server maximum capacity  $c_i^m$ . The quadratic constraint 1.3 means that the link capacity of each server (defined as  $d_i^m$ ) should not be exceeded by the data exchanged between the VEs on that servers and the ones on other servers.  $d_{j_1, j_2}$  indicates the data exchanged between the VEs  $j_1$  and  $j_2$ , while the formula  $x_{i, j_1}(t) + x_{i, j_2}(t) - 2x_{i, j_1}(t)x_{i, j_2}(t)$  corresponds to the binary operator based formulation  $x_{i, j_1}(t) \oplus x_{i, j_2}(t)$ : since VEs that are placed on the same physical server do not use network links to exchange data, our model only takes into account the VEs located on different servers. The constraint 1.4 requires that the available memory on each servers  $m_i^m$  is not exceeded by the sum of memory demands  $m_j(t)$  of the VEs allocated on that server. The constraint 1.5 indicates that each VE must be allocated on one and only one server (classic bin-packing problem constraint). The following constraints concern the VEs migration process. The short notation used in constraint 1.6 actually combines two different constraints: first, each VE cannot be involved in more than one migration (inequality constraint); second, a VE involved in a migration must appear in both the matrices  $g_{i, j}^-(t)$  and  $g_{i, j}^+(t)$ . The constraint 1.7 indicates that a VE migration must start from the server where the VE was allocated at the previous time slot ( $t - 1$ ), while the constraint 1.8 requires that a VE migrates only toward a server where the VE is allocated at time  $t$  (we add this constraint for the clarity of the model even if it is actually redundant, because satisfied by constraint 1.9). The constraint 1.9 means that the VEs allocation at time  $t$  must be the result of the allocation at the previous time  $t - 1$  plus the VEs migrations. Finally, the constraint 1.10 expresses the boolean nature of  $x_{i, j}(t)$ ,  $g_{i, j}^+(t)$ ,  $g_{i, j}^-(t)$ , and  $O_i(t)$ .

## 5 Case study

We now introduce a case study showing how the proposed model can be applied to a SDDC with Virtual Machines (VMs) and Virtual Routers (VRs), in short referred to as Virtual Elements (VEs). The present section starts with a description of the considered case study; next, we provide a performance comparison between the proposed VEs allocation model and other alternatives presented in literature. A qualifying point of our analysis is considering the contribution of VEs migration on the overall energy consumption of the data center. Furthermore, we discuss how the size of the data center impacts on the optimization model performance.

### 5.1 Experimental setup

Let us start with the experimental setup of the considered case study. We consider a time slot with a duration  $\mathcal{T}=15$  minutes for the migration of VEs among the infrastructure. Each VE is implemented as a Virtual Machine (that my host routing software in the

case of a VR) that requires 4 cores and 40 GB of RAM. For the server we rely on the data provided by the the energystar datasheets<sup>4</sup> that are freely available. In particular, we focus on a Dell R410 server with a  $2 \times 6$  cores Xeon X5670 2.93MHz and 128 GB of RAM, so that each server can host up to three VMs. For the considered servers, the power consumption ranges from 197.6 W to 328.2 W.

For the data center network, we recall that its architecture is based on a two-level topology as described in Fig. 2. In our case study we assume communication between VEs not involving data exchange across different Pods consumes half energy with respect to a communication passing both levels of the data center network. Clearly, communication between VEs on the same server has no cost associated. The model for network communication energy cost is obtained by combining multiple sources. On one hand, the switching infrastructure energy cost is based on the Cisco Catalyst 2960 series data sheet<sup>5</sup>. On the other hand, we rely on technical blogs<sup>6</sup> to infer the power consumption when a link can take advantage from *idle mode* and other advanced energy saving features. In the case study we consider that the per-port network power is 4.2 W, while the energy cost for transferring one byte of data is 3 or 6 mJ in the case communication occurs within one Pod or across multiple pods, respectively. It is worth to note that, even if in our case study we consider an homogeneous data center, the model can capture much more complex and heterogeneous scenarios.

Finally, we use the IBM ILOG CPLEX 12.6<sup>7</sup> solver that implements solution algorithms that can cope with the non-convex and quadratic nature of our problem.

In our case study we rely on time series describing the resource (CPU, memory, and network) utilization of VMs in a private cloud data center hosting a e-health application. We also assume that VRs, implemented through Additional VMs, are part of the virtual networks hosted in the data center and we consider the resource consumption on these VRs as proportional to the network traffic. All the VEs show resource utilization characterized by regular daily patterns. In our case study we consider, by default, 80 VEs, resulting in 20-30 servers being powered on. While this scenario is small compared to large public cloud data centers, we consider that for our goal, that is the validation of the optimization model proposed in the present chapter, the case study is significant. Furthermore, we consider that the scalability of the VEs allocation process may be improved integrating a Class-based approach as described in [3, 21], where the solution of a small allocation problems can be replicated as a building block of a larger solution.

In our case study we compare three different models for VEs allocation. The first model, namely *Migration-Aware (MA)*, is the model proposed in our study and detailed in Section 4. The second model, namely *No Migration-Aware (NMA)*, differs from our proposal because it does not consider the cost for VEs migration. For the sake of our model, this means that we consider  $\mathcal{E}_{M_j}(t) = 0 \forall j \in \mathcal{N}$  in the objective function Eq. 1.1. The NMA model is consistent with other proposals in literature, such as [22]. The third and last model, namely *No Network-Aware (NNA)*, does not consider neither

<sup>4</sup> [https://www.energystar.gov/index.cfm?c=archives.enterprise\\_servers](https://www.energystar.gov/index.cfm?c=archives.enterprise_servers)

<sup>5</sup> [http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-2960-x-series-switches/data\\_sheet\\_c78-728232.html](http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-2960-x-series-switches/data_sheet_c78-728232.html)

<sup>6</sup> <http://blogs.cisco.com/enterprise/reduce-switch-power-consumption-by-up-to-80>

<sup>7</sup> [www.ibm.com/software/commerce/optimization/cplex-optimizer/](http://www.ibm.com/software/commerce/optimization/cplex-optimizer/)

migration nor network-related energy costs. The model just minimize the number of powered-on servers, as in [2]. The last model (NNA) and, to a lesser degree, the second one (NMA) are more suitable for the traditional data centers, where network and management are considered as less important issues, while most focus is devoted to the management of computational demands of VEs (in a traditional scenario, VEs consist only in VMs).

A critical point in the traces for network traffic available to us is that we do not have a full description of the data exchange between each couple of VEs and we limit our knowledge to the total amount of data coming in/out form each single VE. To cope with this limit, we reconstruct the data exchange between each couple of VEs creating two different scenarios: *Network 1* and *Network 2*. In the Network 1 scenario we randomly distribute among the VEs the incoming/outgoing traffic to/from each VEs making sure that the total traffic still matches. In the Network 2 scenario we consider in the traffic distribution the presence of a Pareto Law, so that 80% of the traffic of each VE goes to just 20% of the remaining VEs. Furthermore, we make sure that the set of VEs with the highest data exchange shifts over time.

In our analysis we consider mainly the total energy  $\mathcal{E}_{tot}$  consumed in the data center. However, we also measure the single energy contributions related to computational demand ( $\mathcal{E}_C$ ), data transfer ( $\mathcal{E}_D$ ), and VE migrations ( $\mathcal{E}_M$ ) to provide additional insight on the model performance.

## 5.2 Model comparison

As a first comparison between the three considered models, we analyze the total energy consumption and its components for the *Network 1* and *Network 2* scenarios. The results are shown in Figure 4 (results in Fig. 4a were already presented in [8]).

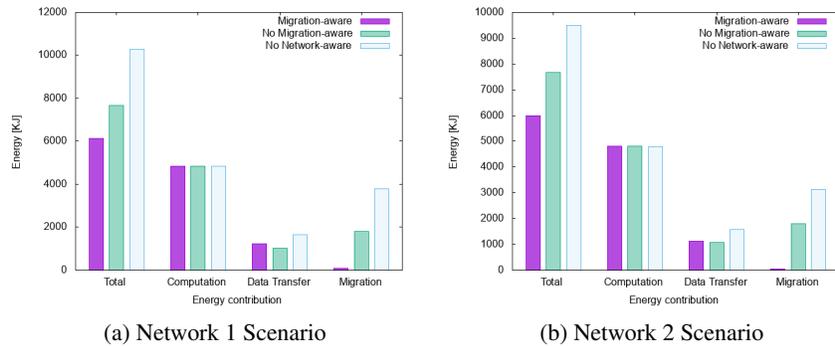


Fig. 4: Energy Consumption Comparison

A first general result is evident from the leftmost column of Fig. 4a and concerns the total energy consumption. The proposed MA model clearly outperforms the alternatives with an energy saving of 20% over the NMA model and up to 40% with respect to the

NNA alternative. To clarify this result, we can compare the contributions to the total energy consumption in the other columns of the figure. From the point of view of energy for computation, all three solution achieve the same results because every approach can consolidate the VEs in the same number of physical servers. The second component of the total energy is related to data transfer. In this case the NMA model provides the best performance due to its objective function that consider just computation and data transfer. As a clarification, it is worth to recall that data transfer does not include the bytes transferred to migrate VEs across the infrastructure. The energy for data transfer in the MA model is 20% higher and the NNA model has an even higher 60% more energy for data transfer. The poor performance of the NNA model is intuitive. However, to understand why the NMA model outperforms the MA alternative we can refer to the last columns of the graphs that is the energy consumed for VEs migration. Specifically, we see that the lower network-related energy consumption of the NMA model comes at the price of a number of migrations that far overweight the benefits of optimized network data exchange. We can thus conclude that not considering the cost of migrations results in an higher than necessary number of migration that, in the end, increase the total energy consumption.

The results for the *Network 2* scenario are shown in figure 4b and confirm the message previously explained for the *Network 1* scenario. Indeed, we observe that the behavior of the different considered models have similar results also under different different network traffic conditions. Furthermore, comparing Fig. 4a and 4b, we observe that even the ratio in energy savings are similar: in the *Network 2* the MA model uses 37% and 22% less energy with respect to the NNA and NMA alternatives, respectively (compared with the 40% and 20% of the *Network 1* scenario).

### 5.3 Impact of migration

We can conclude, from the previous set of experiments that the network-aware models (MA and NMA) clearly outperform the NNA alternative in reducing the energy consumption in a modern data center. It is then interesting to delve into a more detailed comparison of MA and NMA models to understand the impact of migration awareness.

Figures 5 and 6 provide a per-time slot breakdown of the energy consumption of the MA and NMA models for the two network scenarios, respectively. Starting with the Network 1 scenario, a comparison of Figures 5a and 5b, provides a further confirmation that the MA model outperforms the NMA alternative not just in terms of global energy consumption over time, but also for almost every time slot, as shown by the total energy consumption (lines with empty squares). We have also a confirmation that the different results in total energy consumption is mainly related to the energy consumption due to VEs migrations, as testified by the similar behavior of the total energy line (empty squares) and migration energy (filled circles). This finding is even more clear if we refer to the Network 2 scenario. Looking at figure 6b (presented also in [8]) we observe that in the NMA model the minimization of data transfer costs (line with empty circles) comes at the cost of a herding effect with burst of migrations that dominates the energy consumption of the data center. Looking at the MA alternative in Figure 6a it is clear that the MA model accepts a sub-optimal energy consumption for data transfer, but

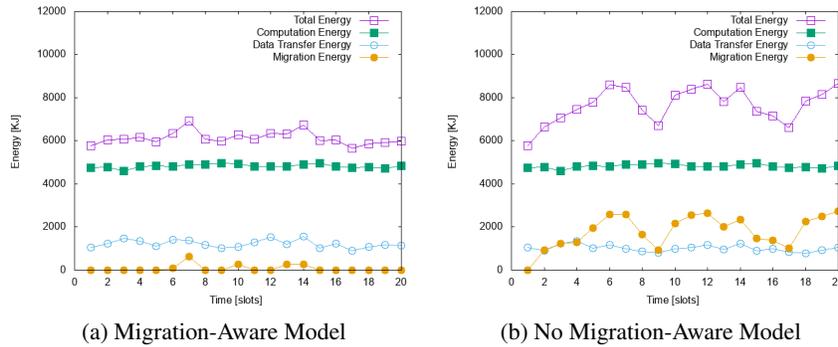


Fig. 5: Energy Consumption Over Time, Network 1 Scenario

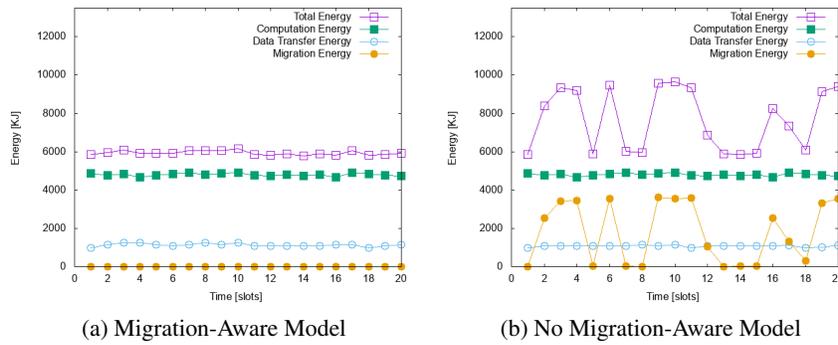


Fig. 6: Energy Consumption Over Time, Network 2 Scenario

achieves a major energy saving by reducing significantly the number of migrations and the associated energy cost compared with the NMA alternative.

#### 5.4 Result stability

The last analysis focuses on the stability of the energy savings with respect to the problem size in terms of VEs.

Figure 7 shows the per-VE energy consumption for the MA model as a function of the data center size with the number of VEs ranging from 20 to 140. We present the analysis for both the Network 1 (Figure 7a, already presented in [8]) and for the Network 2 scenarios (Figure 7b). The graphs for both scenarios, demonstrate that the per-VE data transfer and migration energy are quite stable with respect to the data center size. On the other hand, the computation energy presents more variability, accounting for the fluctuations in the total energy.

The variability in the computation energy is explained by the impact of fragmentation on the server consolidation process. Especially for small problem sizes, we may have servers not fully utilized and, due to the small problem size the cost for a

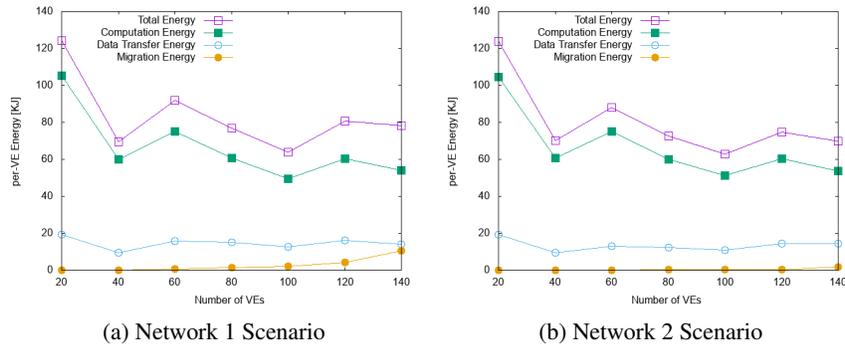


Fig. 7: Energy Consumption vs. Problem Size

marginally utilized server (dominated by the idle power of the server) is spread over a limited number of VEs, thus determining a low efficiency in the server consolidation. The fragmentation effect is amplified by adopting the Class-based consolidation model [3], that trade the possibility of sub-optimal allocations for scalability in the VEs allocation. However, as the data center grows the fraction of servers under-utilized due to fragmentation in the VEs allocation is reduced, resulting in more stable performance with problem size.

## 6 Conclusions and Future Work

In this chapter we discussed the problem of energy-wise optimization of VEs allocation in cloud data centers. This discussion may be applied to existing data centers, but it is mainly focused to a new generation of infrastructures which follow the paradigm of a Software-Defined Data Center, leveraging technologies such as network virtualization and software-defined networks.

Our main contribution is to point out the key differences of these new type of data centers with respect to traditional scenarios and to propose an optimization model to determine VEs allocation in order to combine three goals. The first goal is the reduction of the number of powered-on physical servers achieving what is called *server consolidation*. The second goal is the reduction of the power consumption for data transfer; as we model the energy cost of data exchange between VEs in the infrastructure, we can reduce this energy component by placing VEs with significant amount of data exchange close to each other (ideally on the same server). The third and last goal is to reduce energy consumption due to VEs migration, where this energy contribution is modeled considering both data transfer and CPU overhead due to migration. The model is designed to automatically evaluate if the cost of migrating a VE is balanced by the benefits of reducing the number of turned on servers and optimizing the data transfer over the data center infrastructure. A qualifying point of our proposal is that the components of the objective function considered in our optimization problem measures directly the energy consumption. As a consequence, we do not need to introduce and tune weight

parameters to merge the often conflicting goals of optimal VEs allocation and of avoiding a high number of VEs migrations, because the components of the objective function can be immediately combined together.

We consider a case study based on traces from a real data center, that confirms the validity of our model. The results show that we can reduce the energy consumption from 60% to 37% with respect to a solution which is not aware of network-related energy consumption, and from 22% to 20% with respect to a model that does not take into account the cost of migrations.

## Acknowledgement

The authors acknowledge the support of the University of Modena and Reggio Emilia through the project *S<sup>2</sup>C: Secure, Software-defined Clouds*.

## References

1. Beloglazov, A., Abawajy, J., Buyya, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems* **28**(5) (2012) 755–768
2. Beloglazov, A., Buyya, R.: Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers. *Concurrency and Computation: Practice and Experience* **24**(13) (September 2012) 1397–1420
3. Canali, C., Lancellotti, R.: Exploiting Classes of Virtual Machines for Scalable IaaS Cloud Management. In: *Proc. of the 4th Symposium on Network Cloud Computing and Applications (NCCA)*. (Jun. 2015)
4. Mastroianni, C., Meo, M., Papuzzo, G.: Probabilistic Consolidation of Virtual Machines in Self-Organizing Cloud Data Centers. *IEEE Transactions on Cloud Computing* **1**(2) (July 2013) 215–228
5. Greenberg, A., Hamilton, J., Maltz, D.A., Patel, P.: The cost of a cloud: Research problems in data center networks. *ACM SIGCOMM Computer Communication Review* **39**(1) (December 2008) 68–73
6. Marotta, A., Avallone, S.: A Simulated Annealing Based Approach for Power Efficient Virtual Machines Consolidation. In: *Proc. of 8th International Conference on Cloud Computing (CLOUD)*, IEEE (2015)
7. Drutskoy, D., Keller, E., Rexford, J.: Scalable network virtualization in software-defined networks. *IEEE Internet Computing* **17**(2) (2013) 20–27
8. Shojafar, M., Canali, C., Lancellotti, R.: A Computation- and Network-Aware Energy Optimization Model for Virtual Machines Allocation. In: *Proc. of International Conference on Cloud Computing and Services Science (CLOSER 2017)*, Porto, Portugal (Apr. 2017)
9. Akyildiz, I.F., Lee, A., Wang, P., Luo, M., Chou, W.: Research Challenges for Traffic Engineering in Software Defined Networks. *IEEE Network* **30**(3) (May 2016) 52–58
10. Eramo, V., Miucci, E., Ammar, M.: Study of reconfiguration cost and energy aware vne policies in cycle-stationary traffic scenarios. *IEEE Journal on Selected Areas in Communications* **34**(5) (2016) 1281–1297
11. Verma, A., Ahuja, P., Neogi, A.: pmapper: power and migration cost aware application placement in virtualized systems. In: *Middleware 2008*. Springer (2008) 243–264

12. Cao, Z., Dong, S.: An energy-aware heuristic framework for virtual machine consolidation in cloud computing. *The Journal of Supercomputing* **69**(1) (2014) 429–451
13. Gu, L., Zeng, D., Guo, S., Ye, B.: Joint optimization of vm placement and request distribution for electricity cost cut in geo-distributed data centers. In: *Computing, Networking and Communications (ICNC), 2015 International Conference on*, IEEE (2015) 717–721
14. Eramo, V., Cianfrani, A., Miucci, E., Listanti, M., Carletti, D., Gentilini, L.: Virtualization and virtual router migration: Application and experimental validation. In: *Proc. of International Teletraffic Congress (ITC)*, IEEE (2014) 1–6
15. Boru, D., Kliazovich, D., Granelli, F., Bouvry, P., Zomaya, A.Y.: Energy-efficient data replication in cloud computing datacenters. *Cluster Computing* **18**(1) (2015) 385–402
16. Yi, Q., Singh, S.: Minimizing energy consumption of fattree data center networks. *SIGMETRICS Performance Evaluation Review* **42**(3) (December 2014) 67–72
17. Mandal, U., Habib, M.F., Zhang, S., Mukherjee, B., Tornatore, M.: Greening the cloud using renewable-energy-aware service migration. *IEEE Network* **27**(6) (November 2013) 36–43
18. Wood, T., Ramakrishnan, K., Hwang, J., Liu, G., Zhang, W.: Toward a software-based network: integrating software defined networking and network function virtualization. *IEEE Network* **29**(3) (2015) 36–41
19. Chiaraviglio, L., Ciullo, D., Mellia, M., Meo, M.: Modeling sleep mode gains in energy-aware networks. *Computer Networks* **57**(15) (2013) 3051–3066
20. Clark, C., Fraser, K., Hand, S., Hansen, J.G., Jul, E., Limpach, C., Pratt, I., Warfield, A.: Live migration of virtual machines. In: *Proc. of 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, USENIX Association (2005)
21. Canali, C., Lancellotti, R.: Scalable and automatic virtual machines placement based on behavioral similarities. *Computing* **99**(6) (June 2017) 575–595
22. Huang, D., Yang, D., Zhang, H., Wu, L.: Energy-aware virtual machine placement in data centers. In: *Proc. of Global Communications Conference (GLOBECOM)*, Anaheim, Ca, USA, IEEE (Dec. 2012)