A Computation- and Network-Aware Energy Optimization Model for Virtual Machines Allocation

Claudia Canali*, Riccardo Lancellotti*, and Mohammad Shojafar†

∗ *Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, Modena, Italy*
{*claudia.canali, riccardo.lancellotti*}*@unimore.it*

† *Italian National Consortium for Telecommunications (CNIT), Rome, Italy*
mshojafar@cnit.it

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

Abstract: Reducing energy consumption in cloud data center is a complex task, where both computation and network related effects must be taken into account. While existing solutions aim to reduce energy consumption considering separately computational and communication contributions, limited attention has been devoted to models integrating both parts. We claim that this lack leads to a sub-optimal management in current cloud data centers, that will be even more evident in future architectures characterized by Software-Defined Network approaches. In this paper, we propose a joint computation-plus-communication model for Virtual Machines (VMs) allocation that minimizes energy consumption in a cloud data center. The contribution of the proposed model is threefold. First, we take into account data traffic exchanges between VMs capturing the heterogeneous connections within the data center network. Second, the energy consumption due to VMs migrations is modeled by considering both data transfer and computational overhead. Third, the proposed VMs allocation process does not rely on weight parameters to combine the two (often conflicting) goals of tightly packing VMs to minimize the number of powered-on servers and of avoiding an excessive number of VM migrations. An extensive set of experiments confirms that our proposal, which considers both computation and communication energy contributions even in the migration process, outperforms other approaches for VMs allocation in terms of energy reduction.

1 Introduction

The problem of reducing the computation-related energy consumption in a Infrastructure as a Service (IaaS) cloud data center is typically addressed through solutions based 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) [Beloglazov et al., 2012, Beloglazov and Buyya, 2012,
Canali and Lancellotti, 2015,
Mastroianni et al., 2013]. However, these solutions typically are not network-aware, meaning that they do not consider the impact of data traffic exchange between the VMs of the cloud infrastructure. Neglecting this information is likely to cause sub-optimal VMs allocation because networks in data centers tend to consume about 10%-20% of energy in normal usage, and may account for up to 50% energy during low loads [Greenberg et al., 2008]. Furthermore, few studies proposing solutions for VMs allocation consider the contribution of VMs migration to energy consumption, both in terms of computational and network costs. For example, the network-aware model for VMs allocation proposed in [Huang et al., 2012] does not consider at all the costs of VMs migration: the VMs allocation for the whole data center is re-computed from scratch every time the model is solved. On the other hand, when VMs migration costs are taken into account, they are usually modeled in a quite straightforward way as in [Marotta and Avallone, 2015]: the allocation model simply takes into account the number of VMs migrations, and relies on parameters (weights) to address the trade-off of minimizing both the number of turned on physical servers and of 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

architectures that join virtualization of computation and communication functions, merging virtual machines, network virtualization and software-defined networks [SDDC-Market, 2016]. In such software-defined data centers, the support for more flexible network reconfiguration allows the migration of both virtual machine and communication channels and virtualized network apparatus [Drutskoy et al., 2013]. Hence, traditional VMs allocation policies, that are network blind or adopt simplified models for migration, will be inadequate to future cloud architectures.

The main contribution of this paper is the proposal of a novel solution to minimize energy consumption in present and future cloud data centers through a computation- and network-aware VMs allocation that also takes into account a detailed model of the energy contributions related to the VMs migration process. 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 VM migration process is modeled to consider both the computational overhead and VM data transfer contributions to energy consumption. Our model exploits a dynamic programming approach where the previous VMs allocation is taken into account to compute the future allocation solution: no external parameters or weights are taken into account to combine the two (often conflicting) goals to reduce the number of powered-on servers and limit the number of VMs migrations.

We evaluate the performance of the proposed model through a set of experiments based on two scenarios characterized by different network traffic exchanges between the VMs of the cloud data center. The results demonstrate how the proposed solution outperforms approaches that do not consider network-aware costs related to data transfer and/or apply simplified models for VMs migration to reduce the energy consumption in cloud data centers. 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 describes the reference scenario for our proposal, while Section 3 describes our model for solving the VMs allocation problem. Section 4 describes the experimental results used to validate our model. Finally, Section 5 discusses the related work and 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. Starting from this scenario, we illustrate the characteristics of the proposed model for energy-efficient management of a cloud data center, focusing on the operations that decide the allocation of the VMs over the physical servers of the infrastructure.

Figure 1 presents the general schema of a networked cloud data center: this schema is suitable to describe both a traditional (currently used) data center and a future data center that rely on software-defined or virtualized network infrastructure. While describing how the proposed solution can be integrated in a traditional infrastructure, we outline that it can be easily applied also to these future software-defined data centers, taking advantage of their characteristics to improve the scalability and performance of the energy-efficient management.

The considered networked 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, while active VMs may be migrated from one server to another one according to the data center management strategies. The VMs are hosted on physical servers, which are grouped into racks. The data center is based on a two-level network architecture, with *Top-of-Rack (ToR) switches* connecting the servers of the same rack, 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 rack (passing through the ToR switch) to different racks (passing through the data center core network).

The utilization of the network is collected by the *network manager* component (in the top-right part of the Fig. 1). In a traditional data center this component is typically dedicated to monitoring functions and VLANs reconfigurations. On the other hand, in a software-defined data center the network manager implements the network control logic (the control plane) for the whole infrastructure, while the actual data collection is implemented at the level of the data plane (for example through counters defined in the Open Flow[1] tables inserted in every network element). It is worth to note that the approach adopted by a software-defined data center makes inherently scalable the operations of network monitoring, differently from what happens in traditional data centers. In both present and future data centers, the information about the network utilization is given as input to the *decision man-*

[1] http://archive.openflow.org/wp/learnmore/

ager of the data center, which is receiving also the monitoring information about the utilization of other VMs resources (e.g. CPU and memory). All the information is collected at the level of the hypervisors located on each physical server; even this process of data collection may be scalable if the system adopts a monitoring mechanism based on classes of VMs that have similar behavior in terms of resource utilization. An example of monitoring solution based on this approach is described in [Canali and Lancellotti, 2012].

The *allocation manager* – on the right side of Fig. 1 – is the data center component responsible for running the model for VMs allocation that determines the optimal placement 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 communication related to the decisions are marked as dashed lines in the Fig. 1). It is worth to note that in the case of a software-defined data center, traffic engineering techniques are typically applied [Akyildiz et al., 2016]. These techniques make possible to perform the data transferring due to the VMs migration (that is, the actual transfer of the VM memory size from the source to the destination server) without affecting the performance of normal (application-related) network traffic. This is another reason that makes the integration of complex resource management policies in a software-defined architecture easy and convenient.

We recall that the allocation manager operates 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 [Beloglazov et al., 2012, Marotta and Avallone, 2015,
Mastroianni et al., 2013]. On the other hand, we consider an approach based on time intervals, where a control of the optimality of the VMs allocation on the physical servers is periodically performed. The main reason for this choice 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 the next section.

3 Problem model

We now introduce the model used to describe the VMs allocation problem that we aim to address in our paper. We recall that the final goal is to optimize the VMs allocation on the data center physical servers in order to reduce the energy consumption due to both the computational and networking processes, including the contributions of VMs migrations, while satisfying the requirements in terms of system resources (that is CPU power, memory, bandwidth) of each VM. We recall that the energy model for VMs migration is an original and qualifying point of our proposal.

3.1 Model overview

We consider a set of servers $\mathcal{M}$, where each server i hosts multiple VMs. Each VM $j \in \mathcal{N}$ has requirements in terms of CPU power c_j, memory m_j and network bandwidth (that we model through the data exchanged between each couple of VMs d_{j_1,j_2}). We assume that the respect of SLA is guaranteed if and only if the VM has its required CPU/memory/network resources (as in [Beloglazov et al., 2012], although in this paper only CPU is considered for the SLA). This SLA is suitable for an Infrastructure as a Service (IaaS) scenario.

Another important point of our model is that time is described as a discrete succession of intervals of length $\mathcal{T}$, differently from other approaches which rely on a reactive model based on events (such as CPU utilization exceeding a given threshold [Beloglazov et al., 2012, Marotta and Avallone, 2015,
Mastroianni et al., 2013]). In our model we consider a generic time interval t, to which VMs resource demands are referred, and we assume to know the situation of the VMs allocation in the data center during the previous time interval $t-1$. Knowledge of previous VMs allocation is required in order to support a dynamic programming approach. The VMs demands for CPU, memory and network during future time slots can be predicted by taking advantage from recurring resource demand patterns within the data centers, such as the cyclo-stationary diurnal patterns typically exhibited by the VMs network traffic [Eramo et al., 2016]. Whenever a new VM enters the system during the time interval t, we place it on the infrastructure using an algorithm such as the Modified Best Fit Decreasing (MBFD) described in [Beloglazov et al., 2012]. For the new VM we assume to have no clear information about its communication with other VMs and about its resource demands, so we can discard inter-VMs

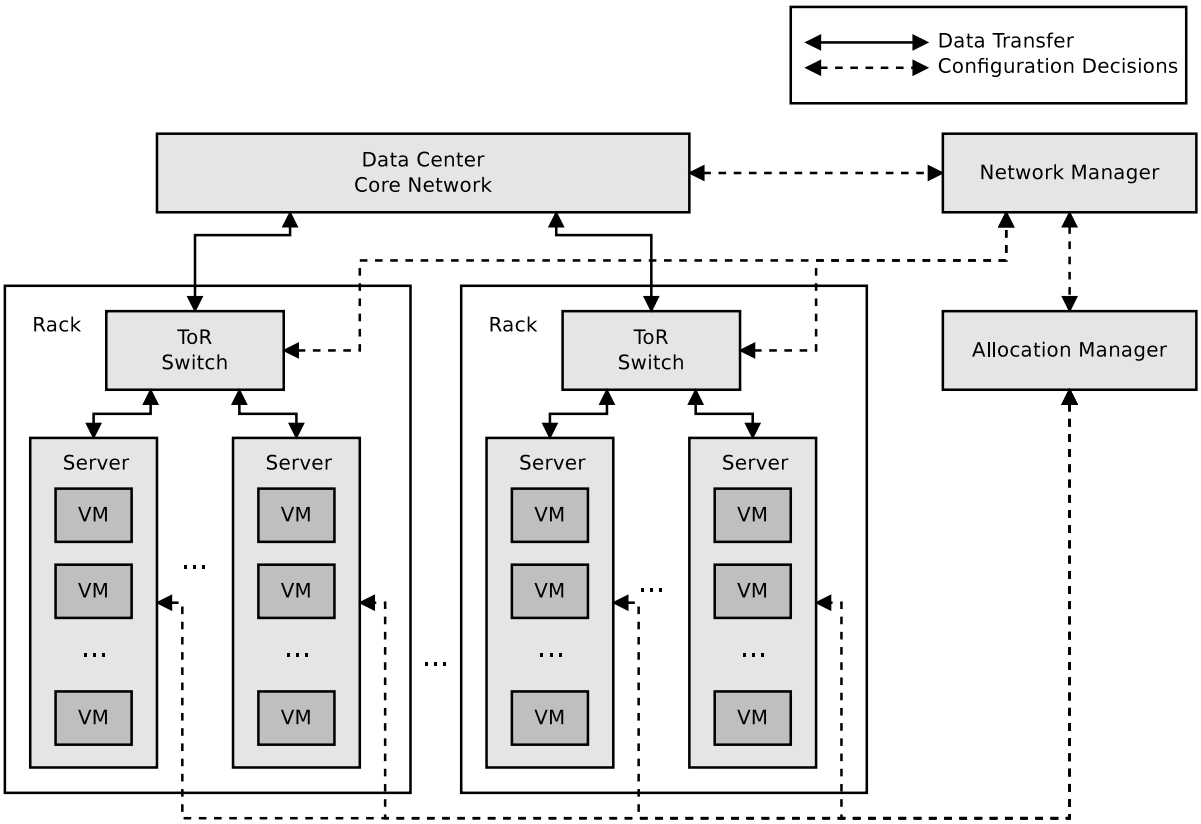


Figure 1: Networked Cloud Data Center

communication costs and we use the nominal values for its resource requirements.

We now describe the optimization model. Table 1 contains the explanation of the main decision variables, model parameters and model internal variables used to describe the considered problem. In our approach, VMs migration represents a way to achieve not only the server consolidation, but also the optimization of the energy consumption due to data transfer between communicating VMs. Similarly to [Marotta and Avallone, 2015], migrations are modeled using 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 VM).

Finally, the decision variables of our problem are: an allocation binary matrix, whose elements $x_{i,j}(t)$ describe the allocation of VM j on server i, and a binary vector whose elements $O_i(t)$ represent the status (ON or OFF) of the physical server i. It is worth to note that the allocation matrix at time $t-1$ represent the actual system status at the *end* of the interval $t-1$, meaning that we remove and add the VMs that left and joined the system during the interval $t-1$.

In the following, we detail the optimization problem that defines the VMs allocation for time interval t.

3.2 Optimization formal model

The formal model of our optimization problem can be described as follows.

Table 1: Notation.

Symbol	Meaning/Role
	Decision variables
$x_{i,j}(t)$	Allocation of VM j on server i at time t
$O_i(t)$	Status (ON or OFF) of server i
	Model parameters
$x_{i,j}(t-1)$	Allocation of VM j on server i at time $t-1$
$\mathcal{T}$	Duration of a time interval
$\mathcal{N}$	Set of existing VMs to deploy $\lvert\mathcal{N}\rvert = N$
$\mathcal{M}$	Set of on servers in the data center $\lvert\mathcal{M}\rvert = M$
$c_j(t)$	Computational demand of VM j at time t
$d_{j_1,j_2}(t)$	Data transfer rate between VM j_1 and j_2 at time t
$m_j(t)$	Memory requirement demand of VM 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
	Model variables
i	Index of a server
j	Index of a VM
$\mathcal{E}_{C_i}(t)$	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 VM j time t
$g_{i,j}^-(t)$	1 if VM j migrates *from* server i time t
$g_{i,j}^+(t)$	1 if VM j migrates *to* server i at time t

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

subject to:

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

$$\sum_{j_1\in\mathcal{N}} \sum_{j_2\in\mathcal{N}} \big(x_{i,j_1}(t) + x_{i,j_2}(t) - 2x_{i,j_1}(t)x_{i,j_2}(t)\big) d_{j_1,j_2}(t) \le$$

$$\le d_i^m O(t), \quad \forall i \in \mathcal{M}, \tag{1.3}$$

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

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

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

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

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

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

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

We now discuss the optimization model in details, staring from the analysis of its objective function (1.1). The VMs allocation process aims to minimize the three main contributions to energy consumption, that are: *Computational demand*, *Data transfer*, and *VM migration*.

The *Computational demand* energy consumption is defined for a generic server i. Its energy consumption is modeled as the sum of two components (as in [Beloglazov et al., 2012]). The first component is the fixed energy cost when the server is ON (power consumption for an idle server is $P_i^m K_{C_i}$). The second component is a variable cost that is linearly proportional to the server utilization, so that the server power consumption is P_i^m when the server is fully utilized. The utilization of a server is obtained using the computational demands of the VMs hosted on that server $c_j(t)$ and the maximum server capacity (c_i^m). The computational demand component can be expressed as:

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

The *Data transfer* is a data center-wise value that corresponds again to the sum of two components, consistently with the model proposed in [Chiaraviglio et al., 2013]. The first component is the power consumption of the idle but turned on network interfaces on the server (defined as P_i^d for server i). The second component is proportional to the amount of data exchanged (based on the parameter d_{j_1,j_2} that describes the data exchange between two VMs j_1 and j_2). It is worth to note that we consider a linear energy model also for the network data transfer, according to [Beloglazov et al., 2012, Chiaraviglio et al., 2013, Eramo et al., 2016]. This model is viable for current data centers and will be even more suitable for future data centers with virtualized and software-defined network functions, where network functions can be considered as abstract computation elements [Drutskoy et al., 2013].

Furthermore, we point out that the matrix $\mathcal{E}_{d_{i_1,i_2}}$ is a square matrix which describes the cost to exchange a unit of data among two different servers and can capture the characteristics of any topology of the data center network in a straightforward way. The global energy cost of data transfer is thus described as:

$$\mathcal{E}_D(t) = \sum_{i\in\mathcal{M}} O_i(t)\mathcal{T}P_i^d +$$

$$+ \sum_{j_1\in\mathcal{N}} \sum_{j_2\in\mathcal{N}} \sum_{i_1\in\mathcal{M}} \sum_{i_2\in\mathcal{M}} x_{i_1,j_1}(t)x_{i_2,j_2}(t)d_{j_1,j_2}(t)\mathcal{T}\mathcal{E}_{d_{i_1,i_2}}$$

The cost of *VMs migration* is a per-VM metric that implements an energy model for the migration

process. When a generic VM j migrates we observe two main effects. First, the whole memory m_j of the VM to be migrated is sent to the destination server (actually the amount of data transferred is slightly higher due to the need to retransmit dirty memory pages, but we neglect this effect due to the typical small size of the active page set with respect to the global memory space of the VM). Second, during the memory copy between two servers, we observe a performance degradation that we quantify using the parameter K_{M_i} for server i hosting VM j. According to the results in literature, this performance degradation is typically in the order of 10% [Clark et al., 2005] but typically takes just a few tens of seconds, which is significantly lower if compared to the time slot duration $\mathcal{T}$. The energy cost for the migration VM j is then:

$$\mathcal{E}_{M_j}(t) = \sum_{i_1 \in \mathcal{M}} \sum_{i_2 \in \mathcal{M}} g^-_{i_1,j}(t) g^+_{i_2,j}(t) \Big(m_j(t) \mathcal{E}_{d_{i_1,i_2}} + \\ + (1 - K_{C_{i_1}}) P^m_{i_1} K_{M_{i_1}} \mathcal{T} + (1 - K_{C_{i_2}}) P^m_{i_2} K_{M_{i_2}} \mathcal{T} \Big)$$

This model is significantly more complex than typical models that just consider the number of migrations such as [Marotta and Avallone, 2015]. However, this complexity is justified by the choice to consider a complete network model in our paper. By adding this component to the objective function, we assume that a migration can be carried out only if the overall cost for migration is compensated by the energy savings due to the better VM allocation.

We now discuss the constraints of the optimization problem. The first group of constraints concerns the capacity limit of the bin-packing problem of VM allocation. Constraint 1.2 means that CPU demands $c_j(t)$ of VMs allocated on each server must not exceed the server maximum capacity c^m_i. The quadratic constraint 1.3 means that for each server, the VM communicating with VMs outside that server must not exceed the server link capacity (defined as d^m_i). The amount of data exchanged between two VMs is d_{j_1,j_2}, and 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 $x_{i,j_1}(t) \oplus x_{i,j_2}(t)$ meaning that we consider just VMs that are allocated on two different servers, because two VMs that are on the same server can communicate without using the resources of the network links. Constraint 1.4 means that memory demands $m_j(t)$ of VMs allocated on each server must not exceed the available memory on the servers m^m_i. Constraint 1.5 is still related to the classic bin-packing problem and means that each VM must be allocated on one and only one server.

The next group of constraints is related to the migration process. Specifically, constraint 1.6 is short notation that combines multiple constraints. First, a VM may be involved in at most one migration (the inequality constraint). Second, a VM involved in a migration must appear in both the matrices $g^-_{i,j}(t)$ and $g^+_{i,j}(t)$. Constraint 1.7 means that a VM may migrate only from a server where the VM was allocated at time $t-1$. Constraint 1.8 means that a VM may migrate only to a server where the VM is allocated at time t (this constraint is redundant, because it is inherently satisfied by constraint 1.9, but we add it for the clarity of the model). Constraint 1.9 expresses how the VM allocation at time t is the result of the allocation at time $t-1$ and of the migrations.

Finally, constraint 1.10 models the boolean nature of $x_{i,j}(t)$, $g^+_{i,j}(t)$, $g^-_{i,j}(t)$, and $O_i(t)$.

4 Experimental results

In this section we present the experimental results performed to validate and evaluate our proposal. After a description of the experimental setup, we compare the performance of the proposed VMs allocation model with other solutions consistent with proposals in literature. Furthermore, we specifically investigate the contribution of VMs migrations to the global data center energy consumption, and evaluate the impact of different data center sizes on the optimization model performance.

4.1 Experimental setup

We now describe the experimental setup considered in our tests.

Server characteristics and power consumption are based on the publicly available energystar datasheets[2]. Specifically we consider a Dell R410 server (power consumption ranges from 197.6 W to 328.2 W) with a 2×6 cores Xeon X5670 2.93MHz ans 128 GB of RAM. We assume that each VM is requiring 4 cores and 40 GB of RAM, so that each server can host up to three VMs. The time slot that we consider has a duration $\mathcal{T}$=15 minutes. As optimization solver we use IBM ILOG CPLEX 12.6[3], that is able to handle the non-convex and quadratic characteristics of our problem.

[2] https://www.energystar.gov/index.cfm?c=archives.enterprise_servers

[3] www.ibm.com/software/commerce/optimization/cplex-optimizer/

To apply the proposed model, we consider traces of resources utilization (CPU, memory, and network) form a real data center hosting a e-health application that is deployed over a private cloud infrastructure; our traces show regular daily patterns. In the data center we consider 80 VMs (as a default value) so that the typical number of physical servers is in the order of 20-30. It is worth to note that this scenario, even if the data center is relatively small, is significant for our goal that is the validation of the proposed optimization model. Moreover, in order to improve the scalability of our approach, we can integrate our model with the Class-based approach to VMs allocation described in [Canali and Lancellotti, 2015, Canali and Lancellotti, 2016], so that we can focus on a small scale allocation problems and then replicate our results on a larger scale.

We recall that the data center network is based on a two-level structure, with Top-of-Rack switches and an upper layer of networking managing the communication among multiple clusters of servers. The information on the network energy consumption is based on the following assumption: the communication between VMs passing through just one level of the network consumes half energy with respect to a communication passing both levels of the data center network. The energy consumption of network apparatus is derived from multiple sources: the basic consumption values for the switching infrastructure of the data center are based on the Cisco Catalyst 2960 series data sheet[4], while the parameters for power reduction when idle mode is used are inferred from technical blogs[5]. From these sources we define the per-port network power as 4.2 W, and the energy cost for transferring one byte of data as 3 mJ in the case we are passing only through the ToR switch and 6 mJ if the core network is involved. It is worth to note that, although our experiments consider an homogeneous data center, the model is much more general and can be directly used to describe a data center where each server and each part of the network infrastructure is characterized by different power consumptions.

Throughout our analysis, we consider three models for VMs allocation, namely *Migration-Aware (MA)*, *No Migration-Aware (NMA)*, and *No Network-Aware (NNA)*. The MA model is our proposal described in Section 3. The NMA model is a model where the cost of VM migration is not considered (that is, we consider $\mathcal{E}_{M_j}(t) = 0 \; \forall j \in \mathcal{N}$ in the objective function). This model is consistent with other proposals in literature, such as [Huang et al., 2012]. Finally, the NNA model does not consider neither migration nor network-related energy costs and only aims to minimize the number of powered-on servers, as in [Beloglazov and Buyya, 2012].

As we do not have a complete definition of the network exchange among the VMs in our traces, but just a description of the global data coming in/out form each single VM (without the breakdown for source or destination), we re-constructed this information by creating two different scenarios. In the first one, namely *Network 1*, we simply randomly distribute the incoming traffic so that the summation remains equal to the available data. In the second scenario, *Network 2*, traffic is randomly distributed across the VMs according to the Pareto law, so that 80% of the traffic of each VM goes to just 20% of the remaining VMs, with the set of VMs with the highest data exchange shifting over time.

The main metric of our analysis to compare the different models is the total energy consumed in the data center ($\mathcal{E}_{tot}$). To provide additional insight on the contributions to the total energy consumption, we also evaluate the single components related to computational demand ($\mathcal{E}_C$), data transfer ($\mathcal{E}_D$), and VM migrations ($\mathcal{E}_M$).

4.2 Model comparison

The first analysis is a comparison of the main models considered in our paper. Figure 2 provides a representation of the total energy consumption and of its components for the three models for the *Network 1* scenario.

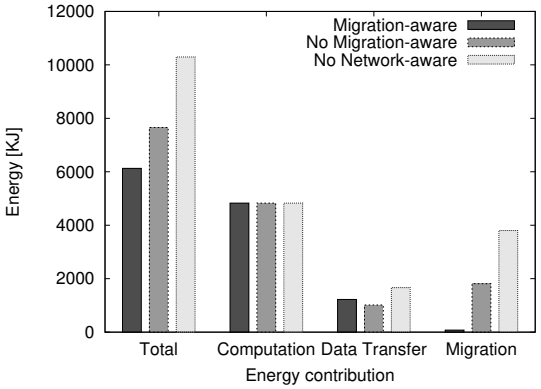


Figure 2: Energy Consumption Comparison

If we focus on the total energy consumption of the three models (leftmost columns of the Fig. 2), we observe clearly that the proposed MA model provides better performance, with an energy saving of 20% over the NMA alternative and up to 40% with

[4]http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-2960-x-series-switches/data_sheet_c78-728232.html

[5]http://blogs.cisco.com/enterprise/reduce-switch-power-consumption-by-up-to-80

respect to the NNA one. The reasons behind this result can be understood when considering the contributions to the total energy in the remaining of the Fig. 2. If we consider just the computation energy contribution, each solution is identical, because every approach can consolidate the VMs in the same number of physical servers. Data transfer is the second source of energy consumption: we observe that the NMA scheme achieves the best results (we recall that the Data Transfer does not include the energy for transferring VMs during the migration, which is included in the Migration contribution). Energy consumption of the NNA model is more than 60% higher and even the MA model show an energy consumption more than 20% higher. The poor performance of the NNA model is intuitive, while to explain the comparison between the MA and NMA models we must refer to the last component, that is the energy consumed for VMs migration. We observe 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. For the NNA approach, again, not considering the cost of migrations leads to a high energy consumption related to this component of the total energy consumption.

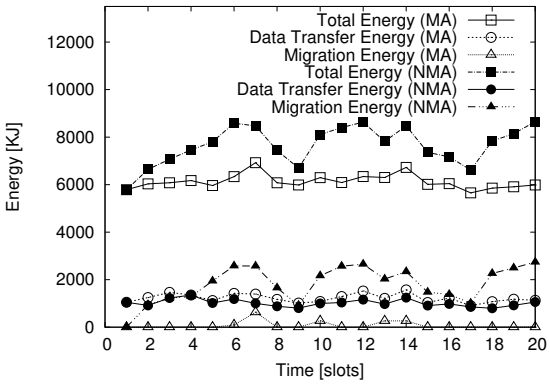
Table 2: Energy Consumption Comparison [KJ]

Model	$\overline{\mathcal{E}}_{tot}$	$\overline{\mathcal{E}}_{C}$	$\overline{\mathcal{E}}_{D}$	$\overline{\mathcal{E}}_{M}$
Network 1				
Migration-aware	6128	4829	1223	76
No Migration-aware	7658	4829	1018	1811
No Network-aware	10297	4829	1664	3803
Network 2				
Migration-aware	5981	4801	1133	47
No Migration-aware	7671	4801	1071	1799
No Network-aware	9511	4799	1572	3140

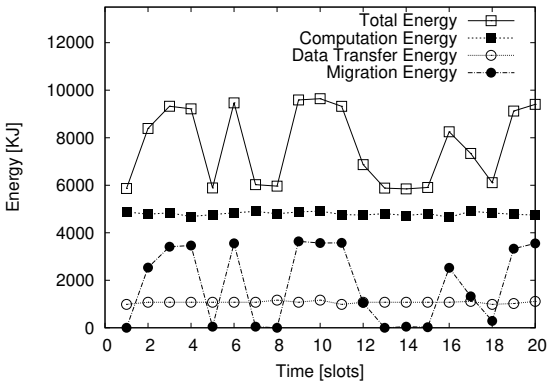
The results for the *Network 2* scenario confirm the message previously explained for the *Network 1* scenario. Table 2 provides results on energy consumption for both network scenarios. If we focus on the second scenario, we observe that all the findings about the energy consumption comparisons are confirmed in this second set of experiments, and also the ratio are similar, with an energy saving of the MA model that is 37% and 22% with respect to the NNA and NMA alternatives, respectively.

4.3 Impact of migration

From the first experiments we have a clear confirmation that network-aware models (MA and NMA) provide a major benefit in terms of energy savings for modern data centers. Hence, we perform a more detailed comparison of the MA and NMA models.



(a) Model Comparison for Network 1 Scenario



(b) No Migration-Aware Model for Network 2 Scenario

Figure 3: Energy Consumption Over Time

Figure 3a compares the per-time slot energy consumption of the data center for the two models. The lines with black and white squares represent the total energy. These lines clearly show how the MA model outperforms the alternative. We also observe that, for every time slot, data transfer-related energy consumption (black and white circles) is lower for the NMA model – this has already been motivated in Section 4.2. The most interesting result is to see how the line shape of total energy cost follows the one of migration, clearly showing the importance of this contribution for the total energy consumption.

Figure 3b shows the per-time slot energy consumption for the NMA model in the case of *Network 2* scenario: the result is a further confirmation of the importance of our choice to add a detailed model for migration-related energy consumption in the proposed energy optimization model. In this case, the variations in data traffic exchange between VMs trigger large burst of migration that account for an energy consumption significantly higher than the energy consumed for data transfer, thus explaining the high total energy consumption. The MA alternative avoids these bursts by limiting substantially the number of migra-

tions, that are carried out only when their overall cost is compensated by the energy savings due to better VMs allocation. As a result, the energy consumption is more stable over time and leads to the major global energy saving already shown in Table 2.

4.4 Result stability

As final analysis, we evaluate if the energy savings of our proposal are stable with respect to the problem size in terms of number of VMs.

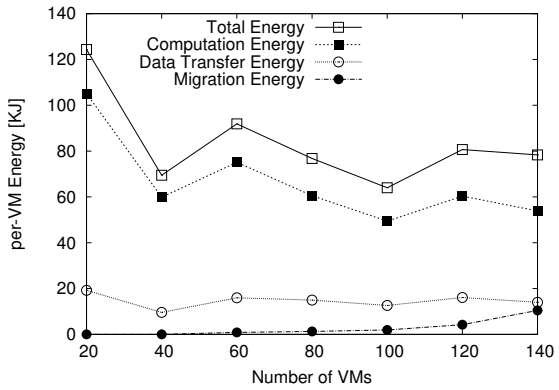


Figure 4: Energy Consumption *vs.* Problem Size

Figure 4 provides an analysis of the per-VM energy consumption as the size of the data center grows from 20 to 140 VMs for the MA model. The graph shows that the per-VM data transfer and migration energy remains rather stable, while the computation energy component is more variable, accounting for the fluctuations in the total energy. This effect is related mainly to the effectiveness of the server consolidation process: depending on the problem size and on the VM packing solutions, we may encounter situations where physical servers are not fully utilized. This fragmentation effect is made more evident by the adoption of the Class-based consolidation model [Canali and Lancellotti, 2015], that increases the VM allocation process scalability at the expenses of possible sub-optimal allocations. We also observe that, as the problem size grows, the general trend of computation energy is towards lower per-VM consumption and reduced fluctuations. Again this can be explained because, as the problem size grows, the quality of the achieved solution for VM allocation increases, because the fraction of servers under-utilized due to fragmentation in the optimization problem solution is reduced.

5 Related work

The problem of VMs allocation has been studied in literature in the last years. One of the most significant papers in the field is [Beloglazov et al., 2012], that defines the VMs allocation problem and proposes some heuristics for its solution. This effort focuses on the placement of new VMs and migration of existing VMs when the server utilization exceeds a specific threshold, with risk of SLA violations. However, the VMs allocation process in [Beloglazov et al., 2012] does not take into account network data exchanges between VMs and focuses only on computational requirements, aiming at minimizing the number of turned on physical servers. A similar approach is considered in [Mastroianni et al., 2013], that proposes a self-organizing and distributed approach for the consolidation of VMs based on two resources, CPU and RAM.

A VMs allocation model taking into account the network-related costs is proposed in [Huang et al., 2012]. However, this study does not model the cost of VMs migration and re-computes the whole VM allocation from scratch every time the model is solved. A similar approach, but aiming at supporting a statistical consolidation of VMs over long periods of time, is proposed in [Wang et al., 2011], where the VM placement problem is modeled as a stochastic bin packing problem. However, this study does not consider the impact of migration as it assumes that VM migration occurs once in very long periods of time (typically in the order of one ore more days). A similar long-term approach is used in [Canali and Lancellotti, 2015, Canali and Lancellotti, 2016]. In this paper, multiple VMs metrics (such as CPU requirements and network utilization) are taken into account in but, as in [Huang et al., 2012, Wang et al., 2011], the migration cost not is considered and a long-term VMs allocation is the main goal of the consolidation bin-packing problem. However, it is worth to note that [Canali and Lancellotti, 2015] proposes a modular approach to the problem of VMs allocation that is applied also to our study in order to improve the scalability of the problem when large data centers are taken into account.

An approach which is closer to our vision is proposed in [Marotta and Avallone, 2015]: this on-line algorithm considers both computational demands and migration costs, following a dynamic programming approach where the previous VMs allocation is taken into account as the basis for the future allocation solution. However, the objective function of the proposed model is quite straightforward from an en-

ergy point of view, as the model simply weights the number of servers turned on against the number of migrations, without providing a detailed model for energy consumption and relying on externally–controlled weights to merge these two sub-objectives.

If we consider explicitly the literature on energy models for cloud data centers, the linear correlation between computational load on the servers and power consumption (the same model we use in our paper) is the most common approach, proposed in [Verma et al., 2008] and adopted also in [Beloglazov et al., 2012, Lee and Zomaya, 2012]. Some alternative non-linear models are adopted in [Boru et al., 2015], but these solutions are typically not very popular in literature, thus justifying our idea to keep as simple as possible the energy model for computation while introducing support for additional energy consumption factors.

The study of networking and data traffic exchange among VMs in a cloud data centers usually is carried out with two possible goals. A first goal is to improve the performance of applications deployed on the data center. For example, in [Meng et al., 2010, Piao and Yan, 2010, Alicherry and Lakshman, 2013] the main goal is to reduce inter-VM communication latency through latency-aware VM allocation policies in order to guarantee fast communication among VMs and fast access to data stored on remote file systems. A similar goal is considered also in the S-CORE system [Tso et al., 2013], but the goal of optimizing performance is relaxed in the simpler objective of avoiding overload on the data center network links. The second goal is related to reduce network-related energy consumption. To this aim, proposals such as [Marotta and Avallone, 2015, Wang et al., 2014] represents first examples of network-aware VMs allocation that aims at reducing energy consumption by placing on the same physical server VMs that have significant data exchange. Another group of papers focused on energy consumption propose some detailed models to capture network-related costs. For example, [Chiaraviglio et al., 2013] introduces a linear power model for energy consumption on a link, while [Chabarek et al., 2008] presents a detailed energy model for the consumption of a router. In [Yi and Singh, 2014] Yi *et al.* present a solution to consolidate traffic into few switches in order to minimize energy consumption in data centers based on a fat-tree network topology: however, the paper solution is dependent on the specific network architecture of the cloud data center and limited effort is devoted to the analysis of computational requirements. Our work clearly fits in the area of research aimed at reducing network-related energy consumption in cloud data centers. In particular, a qualifying point of our contribution is adopting a sophisticated energy models for data exchange and using it to propose an innovative computation- and network-aware model for VMs allocation.

6 Conclusions and Future Work

Throughout this paper we tackled the problem of energy-wise optimization of VMs allocation in cloud data centers. Our focus encompasses both traditional and future software-defined data centers that leverages technologies such as network virtualization and software-defined networks.

Specifically, we proposed an optimization model to determine VMs allocation in order to combines three goals. First, consolidation of VMs aims to reduce the number of powered-on physical servers. Second, the model considers data exchange between VMs and aims to reduce power consumption for data transfer by placing VMs with significant amount of data exchange close to each other (ideally on the same server). Finally, we also model energy consumption due to VMs migration considering both data transfer and CPU overhead due to this task. This model allows to easily evaluate if the cost of migrating a VM is balanced by the benefits of reducing the number of turned on servers and optimizing the data transfer over the data center infrastructure. It is important to note that the components of the objective function of our optimization problem measures directly the energy consumption and can be immediately combined without the need to add weight parameters to merge the (often conflicting) goals of optimal VMs allocation and of avoiding a high number of VM migrations.

Our experiments, based on traces from a real data center, confirm the validity of our model, that reduces 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.

This paper is just a first step of a research line that aims to provide innovative solutions for the energy management of software-defined data centers. Future efforts will focus mainly on improving the scalability of our approach through the proposal of heuristics for determining the VMs migration strategy, with a specific focus on the most advanced features of software-defined infrastructures for the management of data exchange among VMs.

Acknowledgement

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

REFERENCES

[Akyildiz et al., 2016] Akyildiz, I. F., Lee, A., Wang, P., Luo, M., and Chou, W. (2016). Research Challenges for Traffic Engineering in Software Defined Networks. *IEEE Network*, 30(3):52–58.

[Alicherry and Lakshman, 2013] Alicherry, M. and Lakshman, T. V. (2013). Optimizing data access latencies in cloud systems by intelligent virtual machine placement. In *Proceedings of IEEE INFOCOM 2013*, pages 647–655.

[Beloglazov et al., 2012] Beloglazov, A., Abawajy, J., and Buyya, R. (2012). Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems*, 28(5):755–768.

[Beloglazov and Buyya, 2012] Beloglazov, A. and Buyya, R. (2012). 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):1397–1420.

[Boru et al., 2015] Boru, D., Kliazovich, D., Granelli, F., Bouvry, P., and Zomaya, A. Y. (2015). Energy-efficient data replication in cloud computing datacenters. *Cluster Computing*, 18(1):385–402.

[Canali and Lancellotti, 2012] Canali, C. and Lancellotti, R. (2012). Automated Clustering of Virtual Machines based on Correlation of Resource Usage. *Communications Software and Systems*, 8(4):102 – 110.

[Canali and Lancellotti, 2015] Canali, C. and Lancellotti, R. (2015). Exploiting Classes of Virtual Machines for Scalable IaaS Cloud Management. In *Proc. of the 4th Symposium on Network Cloud Computing and Applications (NCCA)*.

[Canali and Lancellotti, 2016] Canali, C. and Lancellotti, R. (2016). Scalable and automatic virtual machines placement based on behavioral similarities. *Computing*, pages 1–21. First Online.

[Chabarek et al., 2008] Chabarek, J., Sommers, J., Barford, P., Estan, C., Tsiang, D., and Wright, S. (2008). Power awareness in network design and routing. In *Proc. of 27th Conference on Computer Communications (INFOCOM)*. IEEE.

[Chiaraviglio et al., 2013] Chiaraviglio, L., Ciullo, D., Mellia, M., and Meo, M. (2013). Modeling sleep mode gains in energy-aware networks. *Computer Networks*, 57(15):3051–3066.

[Clark et al., 2005] Clark, C., Fraser, K., Hand, S., Hansen, J. G., Jul, E., Limpach, C., Pratt, I., and Warfield, A. (2005). Live migration of virtual machines. In *Proc. of 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*. USENIX Association.

[Drutskoy et al., 2013] Drutskoy, D., Keller, E., and Rexford, J. (2013). Scalable network virtualization in software-defined networks. *IEEE Internet Computing*, 17(2):20–27.

[Eramo et al., 2016] Eramo, V., Miucci, E., and Ammar, M. (2016). Study of reconfiguration cost and energy aware vne policies in cycle-stationary traffic scenarios. *IEEE Journal on Selected Areas in Communications*, 34(5):1281–1297.

[Greenberg et al., 2008] Greenberg, A., Hamilton, J., Maltz, D. A., and Patel, P. (2008). The cost of a cloud: Research problems in data center networks. *ACM SIGCOMM Computer Communication Review*, 39(1):68–73.

[Huang et al., 2012] Huang, D., Yang, D., Zhang, H., and Wu, L. (2012). Energy-aware virtual machine placement in data centers. In *Proc. of Global Communications Conference (GLOBECOM)*, Anaheim, Ca, USA. IEEE.

[Lee and Zomaya, 2012] Lee, Y. C. and Zomaya, A. Y. (2012). Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing*, 60(2):268–280.

[Marotta and Avallone, 2015] Marotta, A. and Avallone, S. (2015). A Simulated Annealing Based Approach for Power Efficient Virtual Machines Consolidation. In *Proc. of 8th International Conference on Cloud Computing (CLOUD)*. IEEE.

[Mastroianni et al., 2013] Mastroianni, C., Meo, M., and Papuzzo, G. (2013). Probabilistic Consolidation of Virtual Machines in Self-Organizing Cloud Data Centers. *IEEE Transactions on Cloud Computing*, 1(2):215–228.

[Meng et al., 2010] Meng, X., Pappas, V., and Zhang, L. (2010). Improving the scalability of data center networks with traffic-aware virtual machine placement. In *Proc. of 29th Conference on Computer Communications (INFOCOM)*. IEEE.

[Piao and Yan, 2010] Piao, J. T. and Yan, J. (2010). A network-aware virtual machine placement and migration approach in cloud computing. In *Proc. of 9th International Conference on Grid and Cooperative Computing (GCC)*. IEEE.

[SDDC-Market, 2016] SDDC-Market (2016). Research and Markets. Software-Defined Data Center (SDDC) Market Global Forecast to 2021. www.researchandmarkets.com/research/grj2gz/softwaredefined.

[Tso et al., 2013] Tso, F. P., Hamilton, G., Oikonomou, K., and Pezaros, D. P. (2013). Implementing scalable, network-aware virtual machine migration for cloud data centers. In *2013 IEEE Sixth International Conference on Cloud Computing*, pages 557–564.

[Verma et al., 2008] Verma, A., Ahuja, P., and Neogi, A. (2008). pmapper: power and migration cost aware application placement in virtualized systems. In *Middleware 2008*, pages 243–264. Springer.

[Wang et al., 2011] Wang, M., Meng, X., and Zhang, L. (2011). Consolidating virtual machines with dynamic bandwidth demand in data centers. In *Proceedings of IEEE INFOCOM 2011*, pages 71–75.

[Wang et al., 2014] Wang, S.-H., Huang, P. P. W., Wen, C. H. P., and Wang, L. C. (2014). Eqvmp: Energy-efficient and qos-aware virtual machine placement for software defined datacenter networks. In *The International Conference on Information Networking 2014 (ICOIN2014)*, pages 220–225.

[Yi and Singh, 2014] Yi, Q. and Singh, S. (2014). Minimizing energy consumption of fattree data center networks. *SIGMETRICS Performance Evaluation Review*, 42(3):67–72.