Multi-Objective Mixed Integer Linear Programming Model for VM Placement to Minimize Resource Wastage in a Heterogeneous Cloud Provider Data Center

Rym REGAIEG, Mohamed KOUBAA, Evans OSEI-OPOKU and Taoufik AGUILI
Laboratoire des Systèmes de Communications
Département des Technologies de l'Information et des Communications
École Nationale d'Ingénieurs de Tunis
Université de Tunis El Manar - BP 37, Le Belvédère - 1002 Tunis - Tunisia
Email: {rym.regaieg,mohamed.koubaa,evans.oseiopoku,taoufik.aguili}@enit.utm.tn

***Abstract*—Virtual Machine Placement (VMP) is one of the pressing issues encountered in cloud computing data centers. VMP is the process of selecting the most suitable physical machine (PM) to host the virtual machines (VMs). Typically, the placement goal falls under one of the following optimization criteria: the maximization of the PM usage or power consumption efficiency. In this paper, we propose a Multi-Objective Mixed Integer Linear Programming model (MOMILP) aiming at simultaneously minimizing the VM rejection ratio, the resource wastage and the number of used PMs. To the best of our knowledge, this is the first work which combines such objectives in a Multi-Objective model. We also assume heterogeneous configuration for the data center which has been proven, through recent research work and industrial experience, to be more cost-effective for some applications especially those with intensive Input/Output (I/O) operations. To assess the performance of the proposed model, a comparative study has been curried out. Through the simulation results, it was observed that the proposed model achieves total gains reaching 35% and 10% in terms of resource wastage and power consumption respectively. It was also reported that the power consumption is not only impacted by the number of used PMs but also by the selected PM configurations.**

I. Introduction

Cloud computing is a processing utility that provides resources to perform complex jobs. Large data centers are used to facilitate the incoming jobs by providing virtual machines characterized by resources such as CPU, memory, disk, etc. Based on the virtualization technology, the Virtual Machines (VMs) are hosted on servers that have enough resource capacities to satisfy their resource requirements. The way to place these VMs into the Physical Machines (PMs) is known as the VM Placement (VMP) problem [1].

As there is a worldwide demand for cloud services, a significant amount of energy is been consumed by the virtualization-based cloud platforms [2]. An outcome of this is that, data centers tend to have high operational (OPEX) cost. In [3], it is reported that the energy consumption of United States Data Centers is about 91 billion kilowatt-hours of electricity in 2013, and is projected to reach 140 billion kilowatt-hours annually by 2020. Therefore, minimizing the energy consumed by a data center will help cloud service providers (CSP) to reduce the energy costs as a major contributor of their total operational costs.

In this context, many VMP approaches have been proposed with respect to specific placement objectives. The objectives can typically fall under a couple of different assumptions such as: the minimization of VM rejection ratio and number of used physical machines [4] or the minimization of energy consumption and network traffic [5] or the minimization of SLA violation and energy consumption [6], ...

This paper proposes a Multi-Objective Mixed Integer Linear Programming model (MOMILP), called Model 1, aiming at simultaneously minimizing the VM rejection ratio, the amount of wasted resources and the number of used PMs. As far as we know, this is the first attempt in which such objectives are combined. In order to ascertain the contribution of the proposed model, a comparative study has been established considering two Integer Linear Programming (ILP) models referred to as Model 2 and Model 3 respectively and taking into account different parameter metrics. As the VMP problem has become a particularly challenging task in non homogeneous hardware infrastructures due to the resource variability of the PMs, a heterogeneous data center is considered for simulations.

The rest of the paper is organized as follows. Section II describes the problem tackled in this paper, Related work is given in Section III. In Section IV, we define the notations used to present the models described in Section V. Section VI shows the experiments evaluating our proposed model and their results. Finally, Section VII concludes the paper.

II. Description of the problem

The VMP process can be stated as follows: given a set of VMs and a set of PMs, both are characterized by multiple dimensional resources (CPU, memory, storage), the VMs should be hosted on the PMs with respect to a given placement goal. Figure 2 shows an example of a VMP process with

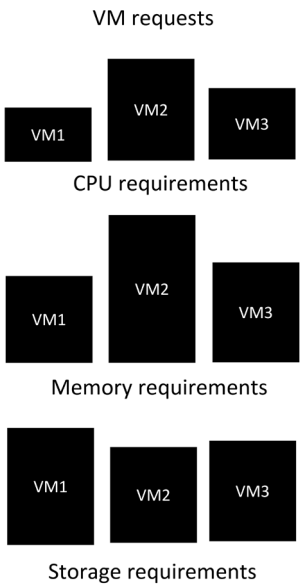


Fig. 1: The VM requests

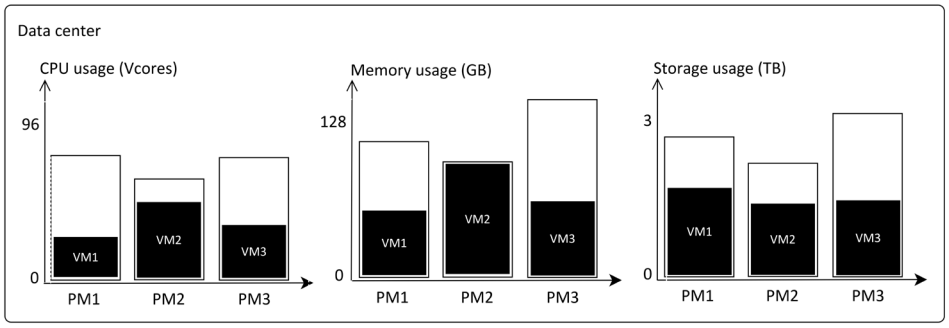


Fig. 2: The VMP without optimizing the resource wastage

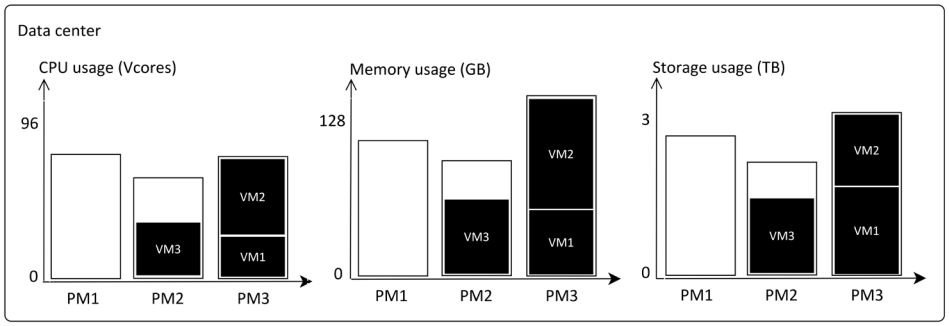


Fig. 3: The VMP taking into account resource wastage optimization

3 VMs and 3 PMs in a heterogeneous data center with an end-goal of maximizing the number of accepted VMs. The VMs requests are given in Figure 1. As it can be seen, after deploying the VMP process, VM_1 is hosted in PM_1, VM_2 is hosted in PM_2 due to the insufficient RAM and disk capacities in PM_1 and VM_3 is hosted in PM_3 as a result of limited resources in both PM_1 and PM_2. Actually, the VMP process produces a large amount of wasted resources due to the underutilization of PMs' resources. As a consequence, an increase in the number of active PMs is noticed, leading to a high power consumption in the data center. In this paper, we look for the optimal VM-PM mapping so that the PMs can be used to their maximum efficiency by minimizing the amount of resource wastage for CPU, RAM and disk across the PMs simultaneously. Additionally, the energy consumption is minimized by hibernating some of the PMs depending on load conditions. Figure 3 shows the expected optimized VMP.

III. Related Work

We here review works which have focused only on the objectives considered in this paper and which have used deterministic algorithms to solve the offline VMP problem. In [7] and [8], Shi, et al., have considered maximizing the CSP revenues, under the placement constraints such as full deployment, security and also resource capacities constraints. An ILP formulation is proposed to compute the exact solution. The proposed VMP approaches are assessed with predefined VMs resource capacities in a homogeneous DC. In [9], Sun, et al., have considered minimizing power consumption, under the PM resource reservation constraints. A matrix transformation algorithm is used to obtain the exact solution. In [10], Ma, et al., have considered minimizing both resource wastage, energy consumption and SLA violation. An ILP formulation is proposed to compute the exact solution. The proposed VMP solutions are evaluated with VMs of customized pattern, i.e., the Cloud user defines the VM resource requirements in a heterogeneous data center [9], [10]. In [11], Xu, et al., have considered minimizing both resource wastage, power consumption, and thermal dissipation costs. An ILP formulation is proposed to compute the exact solution.

Most of the studied works have computed the resource wastage only over CPU and RAM resource dimensions. However, the problem of resource wastage in light of CPU, RAM and disk has not been investigated. Moreover, as far as we know there are no comparative studies between deterministic models. Therefore, the proposed model is convenient and imperative for the fast development of the cloud computing environments.

IV. Notations

We use the following notations and typographical conventions:

Index conventions

- i and j as subscript usually denotes a virtual machine request and a physical machine index respectively.

The parameters

- N corresponds to the number of virtual machines arriving at the Data Center to be hosted. The VM request numbered i, denoted v_i, $\forall\ 1 \leq i \leq N$, is defined by

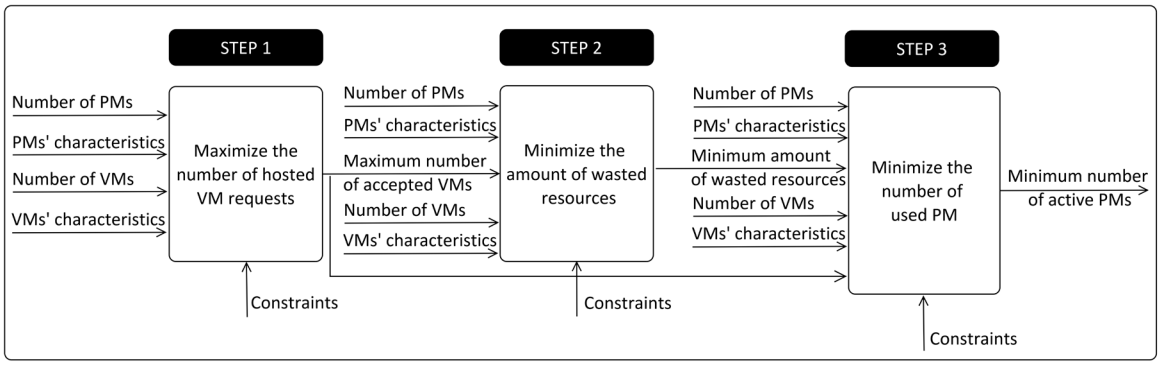


Fig. 4: The Multi-Objective Mixed ILP model

the tri-tuple (c_i, r_i, s_i) where c_i, r_i and s_i are the CPU, memory and storage requirements of VM v_i.

- M corresponds to the number of physical machines in the Data Center. The PM numbered j, denoted P_j, $\forall$ $1 \leq j \leq M$ is characterized by the tri-tuple (C_j, R_j, S_j) where C_j, R_j and S_j are the CPU, memory and storage capacities of PM P_j.

The variables

- The binary variable λ_{ij}. λ_{ij}=1 if the VM v_i is hosted by the physical machine P_j. λ_{ij}=0, otherwise.
- The binary variable ϕ_j. ϕ_j=1, if there is at least one virtual machine hosted by physical machine P_j. ϕ_j=0, otherwise.

V. The model

A. The Multi-Objective Mixed Integer Linear Programming Model

The Multi-Objective Mixed ILP model (Model 1) relies on three separate steps to compute the optimal VM-PM mapping, as shown in Figure 4. Using the previous notations, Step 1, Step 2 and Step 3 are given in Table I.

Step 1 computes the VM-PM mapping with the objective of maximizing ψ_{max}, the number of hosted VM requests. Equations (2) ensures that each VM request v_i is hosted by at most one physical machine P_j. Equations (3) ensures that the total amount of CPU consumed by the VMs hosted at a PM P_j is at most equal to the total amount of CPU available at PM P_j, C_j. Equations (4) and (5) are roughly similar to (3) in that, the CPU resource is replaced by both the memory and storage resources respectively. Equations (6) ensures that λ_{ij} variables are binary.

It may happen that multiple VM-PM mapping solutions exist for the same number of rejected VM requests. Step 2 selects a solution that, in addition, minimizes the resource wastage, δ_{min}. The amount of wasted resources is computed as total amount of unused resources on active PMs [11]. Equations (8) ensures that the number of accepted VM requests must be at least ψ_{max}, computed by Step 1. Equations (9) and (10) define ϕ_j variables. Equations (11) ensures that ϕ_j variables are binary.

TABLE I: Model 1

Step 1

Given N, M, C_j, R_j, S_j, c_i, r_i and s_i

Maximize $$\psi_{max} = \sum_{i=1}^{N}\sum_{j=1}^{M}\lambda_{ij} \quad (1)$$

$$\sum_{j=1}^{M}\lambda_{ij} \leq 1, \qquad \forall 1 \leq i \leq N \quad (2)$$

$$\sum_{i=1}^{N} c_i\lambda_{ij} \leq C_j, \qquad \forall 1 \leq j \leq M \quad (3)$$

$$\sum_{i=1}^{N} r_i\lambda_{ij} \leq R_j, \qquad \forall 1 \leq j \leq M \quad (4)$$

$$\sum_{i=1}^{N} s_i\lambda_{ij} \leq S_j, \qquad \forall 1 \leq j \leq M \quad (5)$$

$$\lambda_{ij} \in \{0,1\}, \qquad \forall 1 \leq i \leq N, \forall 1 \leq j \leq M \quad (6)$$

Step 2

Given N, M, D, C_j, R_j, S_j, c_i, r_i, s_i and ψ_{max}

Minimize

$$\delta_{min} = \sum_{j=1}^{M}\left(3 - \left(\sum_{i=1}^{N}\left(\frac{c_j\lambda_{ij}}{C_j} + \frac{r_j\lambda_{ij}}{R_j} + \frac{s_j\lambda_{ij}}{S_j}\right)\right)\right) - 3\left(M - \sum_{j=1}^{M}\phi_j\right) \quad (7)$$

$$\psi_{max} \leq \sum_{i=1}^{N}\sum_{j=1}^{M}\lambda_{ij} \quad (8)$$

$$\lambda_{ij} \leq \phi_j, \qquad \forall 1 \leq i \leq N, \forall 1 \leq j \leq M \quad (9)$$

$$\phi_j \leq \sum_{i=1}^{N}\lambda_{ij}, \qquad \forall 1 \leq j \leq M \quad (10)$$

(2), (3), (4), (5) and (6)

$$\phi_j \in \{0,1\}, \qquad \forall 1 \leq j \leq M \quad (11)$$

Step 3

Given N, M, D, C_j, R_j, S_j, c_i, r_i, s_i, ψ_{max} and δ_{min}

Minimize $$\theta_{min} = \sum_{j=1}^{M}\phi_j \quad (12)$$

$$\sum_{j=1}^{M}\left(3 - \left(\sum_{i=1}^{N}\left(\frac{c_j\lambda_{ij}}{C_j} + \frac{r_j\lambda_{ij}}{R_j} + \frac{s_j\lambda_{ij}}{S_j}\right)\right)\right) - 3\left(M - \sum_{j=1}^{M}\phi_j\right) \leq \delta_{min} \quad (13)$$

(2), (3), (4), (5), (6), (8), (9), (10) and (11)

Once again, many possible solutions may exist at the end of Step 2, the last (Step 3) selects among the possible ones, the solution that, in addition, minimizes the number of used PMs, θ_{min}. Equations (13) ensure that the total amount of wasted resources in the DC must be at most δ_{min}, computed by Step 2.

In order to assess the performance of Model 1, we propose to compare its results to those of two models described in the following subsections.

B. Model 2

Model 2 is a Two-Objective ILP model that relies on two separate steps to compute the optimal VM-PM mapping. Step 1 is the same as Model 1 with the objective of maximizing the number of hosted VM requests, ψ_{max}. It may happen that multiple VM-PM mapping solutions exist for the same number of rejected VMs. Step 2, given in Table II, selects the solution that, in addition, minimizes the number of used PMs, θ_{min}.

TABLE II: Model 2 - Step 2

Given N, M, C_j, R_j, S_j, c_i, r_i, s_i and ψ_{max}

Minimize $\theta_{min} = \sum_{j=1}^{M} \phi_j$ (14)

(2), (3), (4), (5), (6), (8), (9), (10) and (11)

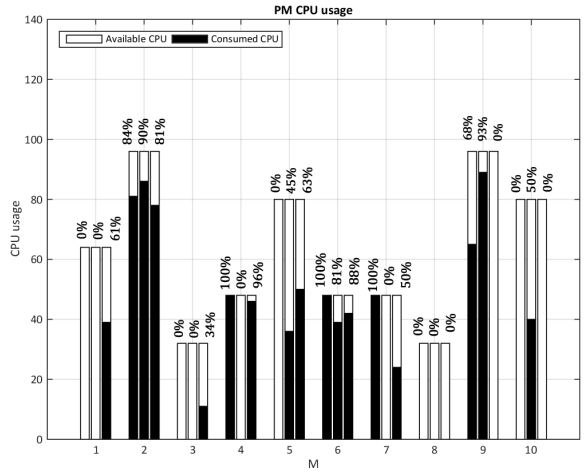
C. Model 3

Model 3 is a Mono-Objective ILP. It computes the optimal VM-PM mapping according to Step 1 of both Model 1 and Model 2.

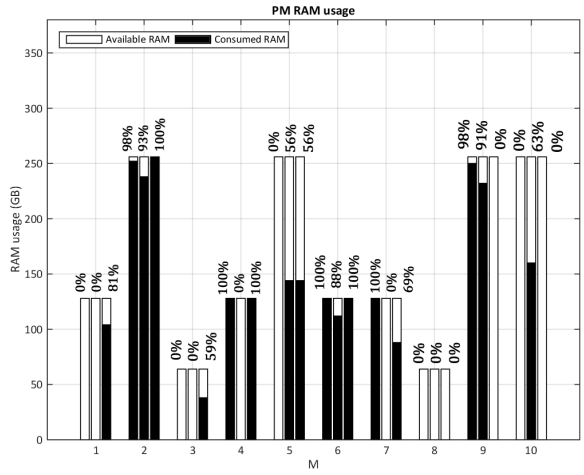
VI. Simulation Results

In this section, we experimentally evaluate and compare the performance of Model 1 to those of Model 2 and Model 3 considering a heterogeneous data center with 10 PMs. Several PM configurations are considered as shown in Table III(a). The number of PMs for each configuration type is also given in Table III(a). We generated 50 test-scenarios, that is, 50 different VM request instances each of which consists of N VM requests generated randomly from a predefined set of VM types referred to as Small (S), Medium (M), Large (L) and XLarge (XL) and whose characteristics are detailed in Table III(b). We used Optimization Programming language (OPL) [12] with CPLEX 12.6.3 [13] to solve the models. The CPLEX solver is run on a windows 10 machine with an Intel Core i7, 2.6 GHz processor and 16GB RAM. The reason why Model 1 cannot solve the VMP problem with over 200 VM requests is due to the NP-hardness of the problem. In the following, each figure shows the simulation results obtained by Model 1, 2 and 3 respectively.

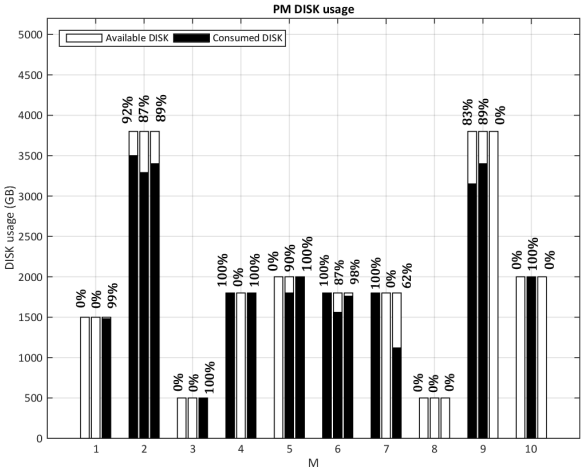
Figures 5(a), 5(b) and 5(c) show the final CPU, RAM and disk usage on each PM as computed by the three models. Each group of three bars plots the resource usage on each PM as computed by Model 1, 2 and 3 respectively. The height of the white bar shows the amount of available resource at



(a) CPUusage



(b) RAM usage



(c) DISK usage

Fig. 5: Average CPU, RAM and Disk usage on each PM, $N = 110$

TABLE III: The PM and VM configurations

(a) The PM configuration

PM	CPU	RAM	DISK	#
C1	32	64	500	2
C2	48	128	1800	3
C3	64	128	1500	1
C4	80	256	2000	2
C5	96	256	3800	2

(b) The VM Configuration

VM	Vcore	Memory(GB)	Disk(GB)
S	1	2	30
M	2	4	60
L	3	8	120
XL	4	16	200

TABLE IV: Average rejection rates of S, M, L and XL VM requests w.r.t. N

N	Average generated S, M, L and XL				Model 1				Model 2				Model 3			
	S	M	L	XL	S	M	L	XL	S	M	L	XL	S	M	L	XL
160	39.28	39.96	40.2	40.56	0.0005	0.001	0.008	0.03	0	0.0005	0.001	0.04	0	0.0005	0.0004	0.04
180	43.56	46.78	44.82	44.84	0	0.005	0.009	0.23	0	0	0.004	0.24	0	0.0004	0.0004	0.24
200	49.3	50.88	50.28	49.54	0	0.003	0.012	0.42	0	0	0.002	0.43	0	0	0.0007	0.44

the PM when no VM are hosted. The height of the black bar shows the amount of the consumed resource after hosting some VMs. The percentage at the top of each bar represents the amount of used resource. One may notice, first of all, that the selected PMs to host the same VM requests instances differ from model to model. One may also notice that the number of used PMs computed by Model 3 is higher to those of Model 1 and Model 2. One may also observe that the amount of the remaining resource across each PM dimension (CPU, RAM and disk) is almost the same for Model 1 compared to the other models. The above results will be explained in details in the subsequent section.

Table IV shows the average rejection ratios for VMs of type S, L, M and XL w.r.t. N. The average rejection ratio for each VM type (T, T = S/M/L/XL) is computed as the number of VM rejected of type (T) to the total number of VM requests of type (T). We notice that the VM requests of type L and XL are the most rejected ones. This result is quite evident regarding L and XL VM requests' resource requirements and the model objective aiming at maximizing the number of hosted VMs.

Figure 6 plots the average resource wastage w.r.t. N. We notice that Model 1 performs efficiently with average gains of 15% and 35% compared to Model 2 and Model 3 respectively. This can be explained by the fact that Model 1 attempts to consolidate the VMs with compatible resource requirements across the PM resource dimensions.

Figure 7 plots the average number of used PMs w.r.t. N. We notice that Model 1 computes an average gain of about 18% compared to Model 3. We also observe that Model 1 and Model 2 perform the same and provide the same result. The last result is due to the VMs' predefined configurations. Using VMs with random configurations should allow Model 1 to compute better results than Model 2.

Figure 8 plots the average total power consumption w.r.t. N. The total energy consumption of the DC is the sum of all the energy consumption of the PMs constituting the DC. The power consumption for a given PM is calculated according to [11]:

$$PC = P_{CPUIDLE} + (P_{CPUMAX} - P_{CPUIDLE})U$$

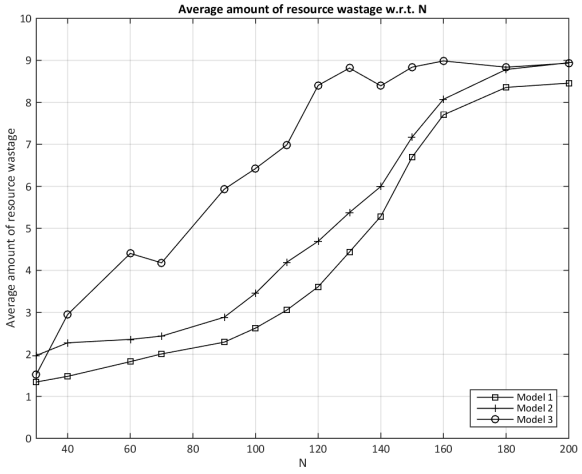


Fig. 6: Average amount of wasted resource w.r.t. N

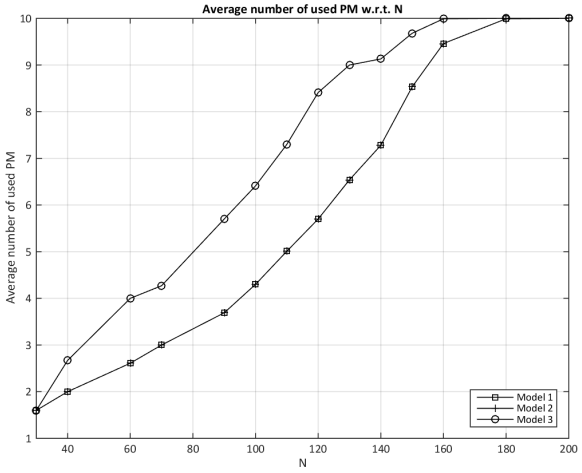


Fig. 7: Average number of used PMs w.r.t. N

where $P_{CPUIDLE}$ and P_{CPUMAX} respectively represent the CPU idle and maximum (100% utilization) of the processor

power, which can be measured by physical meter and U denotes the CPU utilization. The $P_{CPUIDLE}$ and P_{CPUMAX} for each PM configuration are given in Table V. One interesting observation from Figure 8 is that Model 1, comparatively, achieves lower power consumption. An average gain of 5% and 10% are observed comparing to Model 2 and Model 3 respectively. This is mainly related to the lower number of used PMs but also to the configuration of the selected PMs to host the VM requests. As shown in Figure 5, for the hosting of 110 VMs, Model 1 uses five PMs whose configurations are C5, C2, C2, C2, C5 whereas Model 2 uses PMs with configurations C5, C4, C2, C5, C4. The configuration of the selected PMs by Model 1 yields a lower power consumption compared to Model 2. For instance, the amount of power consumption resulting from 10% of CPU utilization of a PM whose configuration is C2 is different from that of a PM with a configuration C4. One may observe that the gain gap between Model 1 and the other models is not too huge for $M = 10$. Such gain value becomes significant in real sized data-centers with a significant number of PMs.

TABLE V: The PM Power Consumption [14]

PM	$P_{CPU-IDLE}$	$P_{CPU-MAX}$
C1	688	936
C2	1032	1404
C3	1376	1872
C4	1720	2340
C5	2064	2808

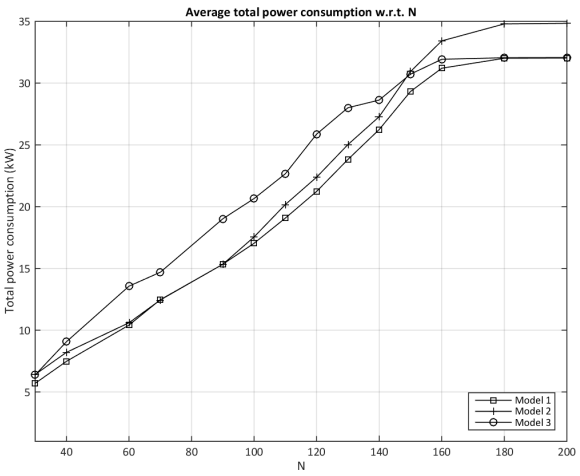


Fig. 8: Average total power consumption w.r.t. N

VII. Conclusion and Future Work

In this paper, we proposed a MOMILP model to address the VM placement problem in CSP data-centers with heterogeneous PM configuration. The objectives are the minimization of the VM rejection ratio, the amount of resource wastage and the number of used PMs. A comparative study has been conducted in order to point out the benefits of the used optimization objectives. Through simulation results, one may notice that Model 1 achieves better performances in terms of resource wastage but especially in terms of power consumption. Future work will focus on how to solve the VMP problem in real sized DC using both heuristics and metaheuristics

References

[1] F. Lopez-Pires and B. Baran. *Virtual Machine Placement Literature*, arXiv preprint arXiv:1506.01509, 2011.

[2] R. Kumar. (2018, May.) *How to measure Energy Consumption in your Data Center, Gartner*, [Online]. Available: http://www.itbriefcase.net/whitepapers/EnergyConsumption.pdf

[3] P. Delforge. (2018, May.) *America's Data Centers Consuming and Wasting Growing Amounts of Energy*, [Online]. Available: https://www.nrdc.org/resources/americas-data-centers-consuming-and-wasting-growing-amounts-energy

[4] B.C. Ribas, R.M. Suguimoto, R.A.N.R. Montano, F. Silva and M. Castilho, *PBFVMC: A New Pseudo-Boolean Formulation to Virtual-Machine Consolidation*, in Brazilian Conference on Intelligent Systems (BRACIS), pp. 201-206, 2013.

[5] M.H. Malekloo, N. Kara, M. El Barachi, *An Energy Efficient and SLA Compliant Approach for Resource Allocation and Consolidation in Cloud Computing Environments*, in journal of Sustainable Computing: Informatics and Systems, vo. 17, pp. 9–24, 2018.

[6] M. Mollamotalebi and S. Hajireza, *Multi-objective dynamic management of virtual machines in cloud environments*, in journal of Cloud Computing, vo. 6, no. 1, pp. 16, 2017.

[7] L. Shi, B. Butler, R. Wang, D. Botvich and B. Jennings, *Optimal placement of virtual machines with different placement constraints in IAAS clouds*, in Symposium on ICT and Energy Efficiency and Workshop on Information Theory and Security, pp. 202-206, 2012.

[8] L. Shi, B. Butler, R. Wang, D. Botvich and B. Jennings, *Provisioning of requests for virtual machine sets with placement constraints in IaaS clouds*, In IFIP/IEEE International Symposium on Integrated Network Management, pp. 499-501, 2013.

[9] M. Sun, W. Gu, X. Zhang, H. Shi and W. Zhang, *A Matrix Transformation Algorithm for Virtual Machine Placement in Cloud*, in IEEE International Conference on Trust, Security and Privacy in Computing and Communications, pp. 1778-1783, 2013.

[10] F. Ma, F. Liu, and Z. Liu, *Multi-objective optimization for initial virtual machine placement in cloud data center*, in journal of Information and Computational Science, vo. 9, no. 16, pp. 5029-5038, 2012.

[11] J. Xu, J.A. B. Fortes, *Multi-Objective Virtual Machine Placement in Virtualized Data Center Environments*, in IEEE/ACM Conference on Cyber, Physical and Social Computing Green Computing and Communications, pp. 179-188, 2010.

[12] *Modeling with OPL* (2018, May.) [Online]. Available: http://www-01.ibm.com

[13] *IBM CPLEX Optimizer* (2018, May.) [Online]. Available: http://www-01.ibm.com

[14] A. Beloglazov, R. Buyya, *Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers*, in Wiley InterScience, 10.1002/cpe.1867, pp. 12,2011.