An Improved Model for Live Migration in Data Centre Simulators

Vincenzo De Maio*, Gabor Kecskemeti†, Radu Prodan*
*Institute of Computer Science, University of Innsbruck
†MTA SZTAKI, Laboratory of Parallel and Distributed Systems
Email: {vincenzo,radu}@dps.uibk.ac.at, kecskemeti.gabor@sztaki.mta.hu

Abstract—Due to the difficulty of employing real data centres' infrastructure for assessing the effectiveness of energy-aware algorithms, many researchers resort on using Cloud simulators. These tools require precise and detailed models for virtualized data centres in order to deliver accurate results. In recent years, many models have been proposed, but most of them either do not consider energy consumption related to virtual machine (VM) migration or ignore some of the energy-impacting components (e.g. CPU, network, storage). In this paper, we propose a new model for data centre energy consumption that takes into account these omitted components. We implement this model in a framework that combines two Cloud simulators: GroudSim that provides the Cloud management side, and DISSECT-CF that provides the internal infrastructure side. We evaluated our model in a comprehensive set of scenarios and obtained an accuracy between 8% and 22% for instantaneous power consumption, and between 8% and 25% for energy consumption.

I. Introduction

Recently, Cloud computing has emerged as a computing paradigm by which computational power is hosted in data centres of specialised providers and rented on-demand to the users based on their occasional needs. Since power consumption has an increased significance for Cloud providers, they are more interested in optimising their data centre's energy efficiency to maximise their profit. One way of improving data centre energy efficiency is to maximise the utilisation of the *physical machines (PMs)*, also referred as hosts, that are often under-utilised according to the study in [1]. For this purpose, data centre operators often apply a technique called workload consolidation that increases the resource utilisation by mapping computational tasks on a subset of the data centre's PMs and shutting down the rest (i.e. putting them in a low power state). Nowadays, computation is mostly running on *virtual machines (VMs)* and thus, this mapping is the mapping of VMs to PMs.

Due to their energy efficiency benefits, a substantial amount of research is currently focusing on workload consolidation algorithms. However, due to the high costs of ownership and expertise for managing such complex infrastructures, it is usually not possible for researchers to use actual data centres for testing the effectiveness of their algorithms. Therefore, there is a growing need for data centre infrastructure simulators that offer an environment which allows the evaluation of various consolidation algorithms. Such simulators need to model the behaviour and the energy consumption of each actor (e.g.

PMs, VMs, routers/switches) and of each activity (e.g. VM migration, PM shutdown/startup) involved in the workload consolidation process. Amongst these activities, *VM migration* is the most important, because it allows moving the state of VMs between PMs, useful for re-mapping VMs according to the data centre's need.

In recent years, several simulators implemented models for VM migration. For example, the work in [2] added a model to the SimGrid [3] simulator that focuses on the migration's performance overhead, but not on its energy consumption.

In this paper, we first propose a new energy consumption model to be used in data centre simulators that considers VM migration operations. Our new approach extends on our previous generic VM migration model [4] implemented in a Cloud infrastructure simulator called DISSECT-CF [5], integrated as a back-end of the user-side GroudSim simulator [6]. Compared to other simulators, our new model increases the accuracy of the simulation of VM migration and similar major activities involved in the workload consolidation process. Our ultimate aim is to provide the distributed systems research community with a model that is easy to implement and able to capture the behaviour of different types of data centre components.

Our migration model is based on the assumption that the source and target PMs are homogeneous, which mimics the current state of most hypervisors (e.g. Xen, KVM) that prevent VM migration between PMs with incompatible architectures. We validated our model by comparing it with real-life measurements from various benchmarks executed on VMs migrated across two different sets of PMs in a private Cloud. Using this strategy, we managed to: (1) improve the energy models of GroudSim/DISSECT-CF, (2) validate our new model's implementation under different kinds of operational scenarios with and without VM migration, and (3) obtain an accuracy between 8% and 22% for instantaneous power, and between 8% and 25% for energy consumption.

The paper is organised as follows. First, we review the related work in Section II. We describe in Section III the implementation of our model in the GroudSim and DISSECT-CF simulators, and evaluate its performance and accuracy in Section IV. Finally, we conclude our paper in Section V.

II. Related work

Several papers proposed models for data centre energy consumptions, however, they either focus on a specific CPU

architecture or assume a linear relationship between CPU usage and energy consumption.

CloudSim [7] is the mostly used and cited simulator of different components of the data centre infrastructure, including internal networking and energy consumption. However, CloudSim assumes that energy consumption exclusively depends on CPU utilization by ignoring other components such as memory and network. Moreover, it does not take into account several important parameters in its VM migration model such as overcommitment and memory dirtying rate.

SimGrid [3] provides a scalable and fast simulation framework of Cloud data centres, Grid and peer-to-peer systems, including a model for simulating VM migration [2]. However, it provides no energy consumption model for VM migration (at the time our paper writing).

GreenCloud [8] offers packet-level simulation for energy-aware Cloud computing data centres, and the capability of separately modelling the energy consumption of all data centre components, including CPU, network, and storage. However, its CPU model is limited to Xeon processors and provides no energy consumption model for live migration.

Our work is based on the GroudSim [6] simulation back-end of the ASKALON system [9] that, due to its integration with the DISSECT-CF [5] Cloud infrastructure simulator, provides models for energy consumption of data centre components, as well as for VM migration and networking.

III. SIMULATION FRAMEWORK

In this section, we describe the simulation framework in which we implemented our models consisting of two parts: GroudSim that provides the user side of the Cloud, and DISSECT-CF that provides the internal Infrastructure as a Service (IaaS) side.

GroudSim is a Java-based simulation toolkit for scientific applications running on Grid and Cloud infrastructures. GroudSim uses a discrete-event simulation toolkit consisting of a future event list and a time advance algorithm offering improved performance and scalability compared to other process-based approaches [7]. GroudSim focuses on the user-side of IaaS Cloud computing and is currently used as an additional back-end in the ASKALON system enabling users to perform seamless development, debugging, simulation and execution of Grid/Cloud applications using the same interface [9]. However, GroudSim lacks knowledge of the internal IaaS Cloud infrastructure. Since this knowledge is essential for the simulation of energy consumption in data centres, we connected it to DISSECT-CF, a compact and highly customisable open source Cloud simulator with special focus on IaaS Cloud systems.

A. VM migration model

In our work, we aim to simulate not only power consumption, but also the VM migration time. VM migration is the process of transferring the VM state from one *source* PM to another *target* PM. We distinguish between two types of VM migration: *non-live* and *live migration*. In the non-live migration, the VM state is transferred after suspending the

VM on the source and then resuming it on the target PM. In the live migration, the state of the VM is transferred while the VM is still running. For both migration types, we identified in [4] three VM migration phases:

- *Initiation phase*, during which the source PM prepares transferring the VM state to the target PM and the target reserves the resources necessary to host the VM;
- *Transfer phase*, during which the VM state is transferred from the source to the target PM in a way depending on whether a non-live or live migration is performed;
- *Activation phase*, during which the source PM frees the resources occupied by the VM and the target starts it.

We therefore define the *VM migration time* $T_{\mathrm{migr}}(v, h, \mathcal{S}, \mathcal{T})$ on PM h for migrating the VM v from the source $\mathcal{S}$ to the target PM $\mathcal{T}$ as the sum of the times required in each phase:

$$T_{\mathrm{migr}}(v, h, \mathcal{S}, \mathcal{T}) = T_{\mathrm{init}}(v, h) + T_{\mathrm{transf}}(v, h, \mathcal{S}, \mathcal{T}) + \\ + T_{\mathrm{activ}}(v, h). \quad (1)$$

In the initiation phase, the source PM prepares a checkpoint of the VM to be sent to the target. In the activation phase, the source PM frees the resource allocated to the VM and the target starts it. Therefore, the times required by both initiation $T_{init}(v, h)$ and activation $T_{activ}(v, h)$ phases are only dependent on the VM size $\mathtt{SIZE}(v)$ and the storage bandwidth on the PM h:

$$T_{\mathrm{init}}(v, h) \quad = \quad T_{\mathrm{activ}}(v, h) \quad = \quad \frac{\mathtt{SIZE}(v)}{\mathtt{BW}_{\mathrm{io}}(h)}. \quad (2)$$

The transfer phase, on the other hand, has a different execution time for a live or a non-live migration. The non-live migration time $T_{\mathrm{transf}}^{\mathrm{nonlive}}$ only depends on the VM size and the bandwidth between the two PMs $\mathtt{BW}_{\mathrm{net}}(\mathcal{S}, \mathcal{T})$:

$$T_{\mathrm{transf}}^{\mathrm{nonlive}} = \frac{\mathtt{SIZE}(v)}{\mathtt{BW}_{\mathrm{net}}(\mathcal{S}, \mathcal{T})}. \quad (3)$$

Instead, live migration is performed iteratively while the VM v is still running. Therefore, the VM state needs to be continuously updated over a predefined number of iterations, set in the hypervisor's configuration. After the initial state transfer, each iteration only transfers the memory pages that have been modified during the previous transfer of the VM state, leading to the following live VM transfer time:

$$T_{\mathrm{transf}}^{\mathrm{live}} = \frac{\mathtt{SIZE}(v)}{\mathtt{BW}_{\mathrm{net}}(\mathcal{S}, \mathcal{T})} + \sum_{i=1}^{\mathcal{I}} \frac{\mathtt{DP}(v, i)}{\mathtt{BW}_{\mathrm{net}}(\mathcal{S}, \mathcal{T})}, \quad (4)$$

where $\mathcal{I}$ is the number of iterations and:

$$\mathtt{DP}(v, i) = \frac{\mathtt{SIZE}(v)}{\mathtt{PS}(v)} \cdot \mathtt{DR}(v, i), \quad (5)$$

represents the number of dirty pages, $DR(v, i)$ is the dirtying rate of the VM v or the percentage of memory pages marked as dirty during an iteration i, and $PS(v)$ is the size of each memory page of VM v.

Section III gives implementation details of this model.
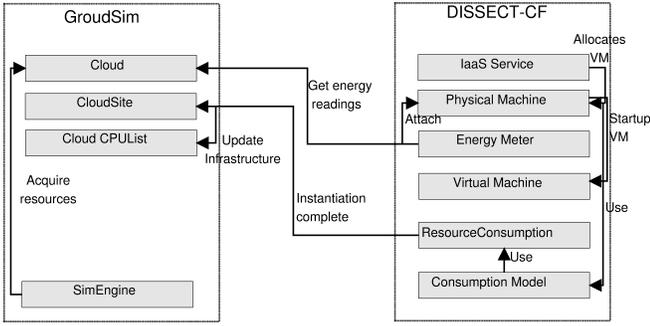
Fig. 1: Interaction between GroudSim and DISSECT-CF.

B. Energy simulation

This section describes the design and implementation of our energy model in the DISSECT-CF infrastructure simulator [5], integrated in the user-oriented GroudSim simulator [6].

1) Energy modelling: In this module, we extended the `ConsumptionModel` class with two subclasses (`CPUConsumptionModel` and `LinearConsumptionModel`). Each class provides an `evaluateConsumption(double load)` method which, when queried, gives the instantaneous power consumption according to the instantaneous load represented by the `load` parameter that models the relative use of the particular resource (e.g. CPU, network, storage). We calculate the load of CPU resources as follows:

$$load_{\text{cpu}} = \frac{\text{CPU}(h,t)}{\text{CPU}^{\max}(h)}, \tag{6}$$

where $\text{CPU}^{\max}(h)$ is the maximum CPU on the PM h and $\text{CPU}(h,t)$ is the load of the PM h at time instance t. We also define the network load as:

$$load_{\text{net}} = \frac{\text{BW}_{\text{net}}(h,t)}{\text{BW}_{\text{net}}^{\max}(h)}, \tag{7}$$

where $\text{BW}_{\text{net}}^{\max}(h)$ is the maximum bandwidth on the network interface of PM h and $\text{BW}_{\text{net}}(h,t)$ is the bandwidth on PM h at the time instance t. Finally, we define the storage load as:

$$load_{\text{io}} = \frac{\text{BW}_{\text{io}}(h,t)}{\text{BW}_{\text{io}}^{\max}(h)}, \tag{8}$$

where $\text{BW}_{\text{io}}^{\max}(h)$ is the maximum storage bandwidth on the PM h and $\text{BW}_{\text{io}}(h,t)$ is the bandwidth at the time instance t.

C. GroudSim and DISSECT-CF Interaction

We display in Figure 1 how to obtain DISSECT-CF energy readings in GroudSim. To measure the energy consumption in a data centre, we need two basic information: the PMs and their load. For this reason, this operation is performed by the *IaaS service* of DISSECT-CF responsible for both instantiating the data centre infrastructure and allocating a VM to a suitable PM. For this purpose, the IaaS service meter attaches an `EnergyMeter` to each PM defined in the data centre. For each PM, we define a `ConsumptionModel` for CPU,

PMs	Available virtual CPUs	Available RAM	Gigabit NIC	Gigabit switch
m01 m02	32 (8 Opteron 8356, dual threaded)	32GB	Broadcom BCM5704	Cisco Catalyst 3750
o1 o2	40 (20 Xeon E5-2690, dual threaded)	128GB	Intel 82574L	HP 1810-8G

TABLE I: Hardware configuration.

network and storage that defines the instantaneous power consumption of each component, as discussed in Section III-B1. Energy meters collect these instantaneous power consumption values with a user-defined frequency and calculate the energy consumption based on the simulated time spent since the last power measurement. At the end of the simulation, the IaaS service's meter aggregates the energy consumption for the entire data centre.

IV. EVALUATION

In this section, we evaluate our simulated energy model for VM migrations by first describing the selected benchmarks and the experimental testbed. Then, we describe how we simulate the execution of these benchmarks on the top of DISSECT-CF. Finally, we compare the results of our simulations with energy traces collected from real executions in the simulated Cloud.

A. Benchmarks

We employed the benchmarks designed in [4] to validate the implementation of our new model of VM migration inside our simulator. We made this choice because: (1) they already proved their effectiveness in testing our VM migration model, and (2) they allow us to check the accuracy of our VM migration model by varying the CPU load and the dirtying rate, which are the parameters that mostly affect the VM migration. We executed each experiment on both sets of PMs described in Table I and running the Xen $4.2.5$ hypervisor. To each PM, we attached an external Voltech PM1000+ power reader to measure its energy consumption. For each experimental run, we start the measurement after deploying the VMs on the PMs, we issue the migration once the power consumption of the PM stabilises, and we end the measurement once the power trace stabilises. We say that the power trace is stable once we obtain twenty consecutive power measurements with a difference lower than the measurement error 0.3%.

B. Simulation benchmarking

After collecting the real world traces, we implemented the same benchmarks on the top of DISSECT-CF to evaluate the accuracy of our simulations. We configured a micro data centre with two PMs matching the configuration of the two kinds of PMs used for regression modelling (see in Table I). We simulated the load by deploying VMs on the PMs and configured each VM to have $4\,\text{GB}$ of memory to resemble the configuration used to build our traces. To simulate the execution of the benchmarks, we assigned computational tasks to each VM resembling the execution of the selected workloads.

PM	MAE-NONLIVE		NRMSE-NONLIVE		MAE-LIVE		NRMSE-LIVE	
	Power [W]	Energy [J]	Power [%]	Energy [%]	Power [W]	Energy [J]	Power [%]	Energy [%]
Source (m01)	42.97	4292.9	16.6	16.9	38.45	5345.8	15.5	8.1
Target (m02)	51.97	4179.5	18.3	15.1	67.33	6341.8	22	9.3
Source (o1)	11.98	2248.6	8.2	14.6	27.6	3375.5	18.4	11
Target (o2)	18.19	2417	14.6	13.2	48.1	5518	14.2	25.6

TABLE II: Error for the each PM set.
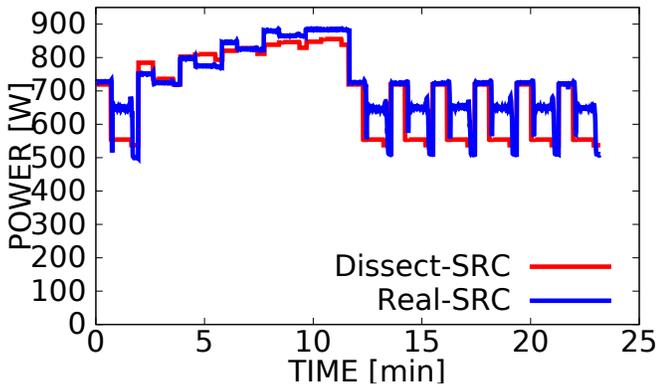
Fig. 2: Results for the (m01, m02) machine set.

C. Simulation results

In this section, we compare the results obtained from our simulator with measurements taken from real experiments. We compute the Mean Absolute Error (MAE) and the Normalised Root Mean Square Error (NRMSE) error metrics on both instantaneous power and energy consumption on both sets of PMs. The results are summarised in Table II for each PM set. We first observe that our simulation is able to provide power values with a MAE below $67.3\,\mathrm{W}$ This value is, however, influenced by the fact that, in some cases, the power consumption is underestimated by around $100\,\mathrm{W}$ like in Figure 2 (between 12 and $24\,\mathrm{min}$). because the test scenarios active during those minutes perform non-live migrations while both PMs are idle. In these situations, the simulator only considers the power consumption caused by the network and I/O operations (added to the idle CPU energy consumption), despite some slight CPU load caused by them too. In future work, we will aim at modelling this inherent CPU load in a generic way to increase the accuracy of the simulator in these unlikely test scenarios too. Nevertheless, NRMSE is in each case between 8% and 22% for instantaneous power consumption, and between 8% and 25% for energy consumption (see Table II), showing that our simulator is able to predict both energy and instantaneous power with good accuracy.

V. Conclusion and Future Work

In this paper, we developed a new energy model for data centre energy consumption that considers CPU, network and storage hardware components. Afterwards, we implemented our model in the user-side GroudSim simulator by exploiting its integration with DISSECT-CF infrastructure simulator. We evaluated the accuracy of our implementation by comparing it with real measurements, showing a NRMSE between 8% and 22% for power prediction and between 8% and 25.6% for energy estimation. In the future, we plan to perform further extensions to our simulator by improving the energy models for network and storage and use them for studying the effects of different energy-aware consolidation algorithms in modern virtualized data centres. We are further interested in validating our simulator with different real-world benchmarks such as TPC-C[1] or SPECPower[2].

References

[1] Luiz André Barroso and Urs Hölzle. The case for energy-proportional computing. *Computer*, 40(12):33–37, December 2007.

[2] Takahiro Hirofuchi, Adrien Lèbre, and Laurent Pouilloux. Adding a live migration model into SimGrid: One more step toward the simulation of infrastructure-as-a-service concerns. In *2013 IEEE 5th International Conference on Cloud Computing Technology and Science*, volume 1 of *CloudCom*, pages 96–103. IEEE, December 2013.

[3] Henri Casanova, Arnaud Legrand, and Martin Quinson. SimGrid: A generic framework for large-scale distributed experiments. In *Tenth International Conference on Computer Modeling and Simulation (UKSIM 2008)*, UKSIM, pages 126–131. IEEE Computer Society, April 2008.

[4] Vincenzo De Maio, Gabor Kecskemeti, and Radu Prodan. A workload-aware energy model for virtual machine migration. In *2015 IEEE International Conference on Cluster Computing*, CLUSTER, pages 274–283. IEEE Computer Society, September 2015.

[5] Gabor Kecskemeti. DISSECT-CF: a simulator to foster energy-aware scheduling in infrastructure clouds. *Simulation Modelling Practice and Theory*, 58(2):188–218, November 2015.

[6] Simon Ostermann, Kassian Plankensteiner, and Radu Prodan. Using a new event-based simulation framework for investigating resource provisioning in clouds. *Scientific Programming*, 19(2-3):161–178, 2011.

[7] Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, César A. F. De Rose, and Rajkumar Buyya. CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 41(1):23–50, January 2011.

[8] Dzmitry Kliazovich, Pascal Bouvry, and Samee Ullah Khan. GreenCloud: a packet-level simulator of energy-aware cloud computing data centers. *The Journal of Supercomputing*, 62(3):1263–1283, December 2012.

[9] Thomas Fahringer, Radu Prodan, Rubing Duan, Jürgen Hofer, Farrukh Nadeem, Francesco Nerieri, Stefan Podlipnig, Jun Qin, Mumtaz Siddiqui, Hong-Linh Truong, Alex Villazón, and Marek Wieczorek. ASKALON: A development and grid computing environment for scientific workflows. In Ian J. Taylor, Ewa Deelman, Dennis B. Gannon, and Matthew Shields, editors, *Workflows for e-Science*, pages 450–471. Springer, 2007.

[1] http://www.tpc.org/tpcc/
[2] https://www.spec.org/benchmarks.html#power