NSysS 2017.
Department of CSE, BUET. 5-8 January, 2017

Cooling Energy Integration in *SimGrid*

A.S.M Rizvi, Tarik Reza Toha, Mohammad Mosiur Rahman Lunar, Muhammad Abdullah Adnan, and A.B.M. Alim Al Islam
Department of CSE, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
Email: rizvi22222@yahoo.com, {tarik.toha, lunar.mosiur, abdullah.adnan}@gmail.com, razi_bd@yahoo.com

***Abstract*—Cooling energy consumption is one of the most significant parts of total energy consumed by distrubuted systems. However, little effort has been spent so far to integrate the cooling energy in simulators that are used for simulating distributed systems. Therefore, in this paper, we propose an integration of cooling energy consumption in a widely-known simulator of distributed systems namely *SimGrid*. Here, we present necessary energy models that are needed to measure cooling energy required for a distributed system. Subsequently, we perform necessary modifications in *SimGrid* to integrate the models. We perform rigorous simulation over different settings using our integrated modules of *SimGrid*. Alongside, we perform real experimentation using settings similar to that used in our simulation. We compare our simulation results against that we find from real experimentation. The comparison reveals applicability of our cooling energy integration in *SimGrid* for simulating diversified distributed systems.**

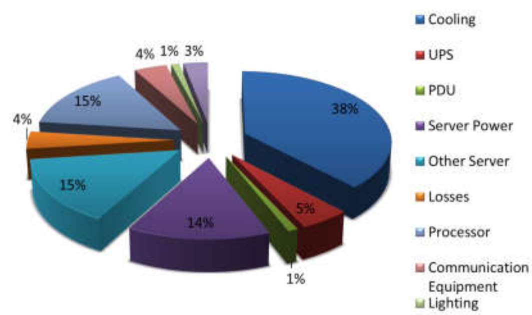


Fig. 1: Energy consumption in data centers

I. Introduction

Distributed computing infrastructures such as data centers consume a considerable part of electricity consumption around the world and their extent of energy consumption is increasing day by day [17]. From Fig. 1, it is evident that around 38% of the total energy consumption of data centers pertains to cooling energy. Therefore, success of any work related to energy consumption of such infrastructures vastly depends on proper realization of the cooling energy. One of the widely adopted methodologies for realizing different aspects of energy consumption is to perform simulation.
Different simulation tools have been developed for computing energy consumption. Among them NS-2 [9] and NS-3 [10] are the two most popular ones. However, NS-2 and NS-3 are yet to offer any support for measuring energy consumption in wired system to the best of our knowledge. Both of those simulators are capable of measuring energy consumption for wireless systems.
SimGrid [2], being a simulation tool developed for simulating distributed systems, offers an energy plug-in for measuring energy consumption in wired systems. However, this energy plug-in is yet to consider cooling energy consumption which is a significant part of energy consumption as mentioned above. Therefore, in this paper, we propose necessary modification in the plug-in to incorporate cooling energy consumption. Here, we integrate cooling energy consumption with the consideration of environment temperature, maximum allowable temperature inside the machine, and the workload of the system under consideration.

We perform validation of our integration through comparing our simulation results against that of real testbed experiments. Based on our work, we make the following contribution in this paper:

- We point out necessary models that need to be incorporated in *SimGrid* to integrate cooling energy consumption in it. Subsequently, we make necessary modifications in the existing *SimGrid* to incorporate the models.
- We simulate different distributed computing systems using our modified *SimGrid* module. Here, we vary both workload and the number of machines.
- We also perform real testbed experiments with similar settings adopted in our simulation. Further, we compare the simulation and testbed results to validate applicability of our proposed modified plug-in of *SimGrid*.

The rest of the paper is organized in following way. In second section we describe our motivation behind the work. Then we gave a small description on the related works. In fourth section our proposed method had been discussed. We validate our methodology in final section with experimental results.

II. Motivation behind the Work

Measuring energy consumption is important for modeling the power distribution and consumption. This is also important for the researchers who want to model energy efficient systems. However, measuring energy consumption of a large system is difficult and troublesome. Hence, there is a necessity of

978-1-5090-3260-0/17/$31.00 © 2017 IEEE

simulating energy consumption of large system. Moreover, sometimes, it is necessary to simulate large-scale experiments, as simulation can give us an easy scalable environment. For example, connecting tens of machines in a laboratory can easily be done however, if we want to work with hundreds of machines, then simulation might be a quick and cost-effective solution. For ad-hoc networks there are some tools such as NS-2, NS-3 etc for measuring the energy consumption. However, a limited number of tools are available for measuring energy consumption in a distributed wired system. *SimGrid* is such a tool which has energy plug-in for simulating distributed wired system. However, *SimGrid* does not have any module for measuring cooling energy. Many a times, cooling energy becomes vital and we cannot neglect it anyway. This is our main motivation behind integrating cooling energy with the contemporary energy module of *SimGrid*.

III. Related Work

Measuring energy consumption has been investigated in various research studies [3], [20], and [19]. However, these studies mostly deals with ad-hoc networks. For example, the study presented in [4] presents an energy consumption model for mobile ad-hoc networks for measuring performance of routing protocols. This study uses NS-2 simulator for measuring energy consumption. Similar other studies [1] also exist in the literature. Besides, some other studies focus on energy efficient protocols in wireless networks [7]. Nonetheless, only a few studies[11] aim at modeling energy consumption in distributed systems. An example of such studies [6] develops different APIs that are used in *SimGrid*. However, to the best of our knowledge, none of these studies has modeled the cooling energy to integrate it within total energy consumption. Also, *SimGrid* energy module has already been used by [14], [12], hence, integrating cooling energy with *SimGrid* is necessary to get more concrete and accurate results. In [5] authors describe a dynamic nature of cooling energy consumption, however, in our case, we try to keep our model simple to make a consistent result with the existing *SimGrid* result. In [13], authors describe in detail about the thermal energy consumption theoretically, our formulation is similar to that in some way, however, due to some constraints of the existing *SimGrid*, we keep our model simpler.

IV. Overview on *SimGrid*

SimGrid [2] is a simulation tool for large distributed system which is written in C language having some Java support. Simulation topology can be created in XML format. Total number of nodes along with the connections among them needs to be assigned to create the topology. In another XML file, deployment scenario of the topology needs to be described. Here, the description of the jobs, their chunk sizes, selection of master and slave nodes need to be described. SimGrid master will distribute the jobs among the workers. Computation time to finish the jobs can be measured in *SimGrid*. Moreover, there is a module to calculate the energy consumption. This energy consumption is proportional to the workload and computation time.

V. Proposed Methodology

As mentioned earlier, using *SimGrid* energy plug-in, we can evaluate all the energy dissipations by the machines of a distributed system. However, this energy plug-in does not include cooling energy consumption. Present energy plug-in uses some easy heuristics to get the energy consumption. This plug-in first evaluates the CPU load, which depends on the task intended to be done by the participant machines. Then it estimates the time necessary to do this task. This time depends on the processing power of the machines. Then, through multiplying CPU load with the required time, *SimGrid* measures the energy consumption for this task.

SimGrid has three types of energy consumption state. One state gets initiated when the CPU is in full-load mode, another state gets initiated when the CPU is in idle mode, and the remaining state gets initiated when the CPU is totally off. All of these states are pertinent to measuring computational power, exhibiting no impact related to cooling power.

To investigate cooling energy in *SimGrid*, we model the cooling energy based on CPU load and temperature difference between environment and maximum allowable temperature of machines. Cooling energy is also dependent on many other factors, however, to correlate with the existing simplified model of energy consumption, we keep our model simple. We have the conventional heat formula as:

$$Q = C_p \times W \times DT \tag{1}$$

where C_p denotes the specific heat. W refers the mass of the airflow per minute, this airflow is necessary to keep the internal temperature of the machines to an allowable temperature, and the mass of the airflow is counted for a particular time, say minute. DT means the change in temperature.
Also, it is not very hard to write:

$$W = CFM \times D \tag{2}$$

Here, CFM denotes cubic feet per minute and D refers to density. We can validate this equation by observing the units. CFM has $volume per minutes$ unit and Density has $mass per volume$ unit. Multiplying we get a $mass per minute$ unit.
Putting the value of Eq. 2 to Eq. 1, we get the following equation:

$$Q = C_p \times CFM \times D \times DT \tag{3}$$

From Eq. 3, we consider the equation for airflow in a chassis as follows:

$$CFM = \frac{Q}{C_p \times D \times DT} \tag{4}$$

Specific heat of a room remains constant and at a certain place where the pressure is constant we can assume density will remain constant. However, these values can be changed at different times of the year. We will consider this variation in our future work. Considering other heat losses through the chassis wall we can write the following formula[15] for

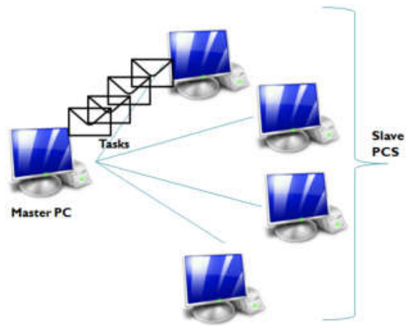


Fig. 2: Master slave relation in our experimental settings

Fahrenheit scale:

$$CFM = \frac{3.16 \times Q}{T_F} \tag{5}$$

Where T_F denotes the maximum allowed temperature in Fahrenheit scale. In Celsius scale it can be written as:

$$CFM = \frac{1.76 \times Q}{T_C} \tag{6}$$

CPU load is responsible for heat generation. If the CPU load is high, a higher amount of heat will be generated. As they show proportional relation, we can write Q as the CPU load instead of heat, at any given time and task. Also, T_C denotes the maximum allowed temperature in Celsius scale.

Algorithm 1: Cooling Power Integration Algorithm

UpdateCoolingEnergy
Input : Work-load, W
Output: Update cooling energy
$T_E \leftarrow EnvironmentTemperature$
$T_A \leftarrow AllowableTemperature$
$T_{Diff} \leftarrow T_E \sim T_A$
$CFM \leftarrow \frac{1.76 \times W}{T_{Diff}}$
$Power \leftarrow PreviousPower + (CFM \times 47.82)$
$ConsumedEnergy \leftarrow$
$Power \times TimeForComputation$

Algorithm 1 shows the overall methodology. Here, environment temperature is T_E and we need to keep the chassis temperature as T_A, which is the allowable temperature. Then the allowed temperature rise will be $T_E \sim T_A$. This difference is the same difference that is showing in Eq. 5 and Eq. 6 as T_F annd T_C. *SimGrid* will also provide the CPU load. Using this CPU load and temperature difference we can get the value of CFM. We use 1 CFM = 47.82 W according to the formula [16] to get the watt value corresponding to the CFM. This is the watt value that is needed to keep the PC in an operative mode. After that, we multiply the watt value with the consumed time. The function of Algorithm 1 will be invoked when the job is finished. The time when the job was started can be retrieved by invoking a *SimGrid* function. We get the current time also to estimate the consumed time to finish the job.

TABLE I: Simulation environment in laboratory

Parameter	Value
No. of master	1
No. of slaves	29
Processor	Intel Core 2 Duo
Processor base frequency	2.4 GHz(3 PCs), 2.66 GHz(10PCs) and 2.8 GHz(17 PCs)
Memory	1GB to 2GB
OS	Ubuntu 14.04 LTS (x86)

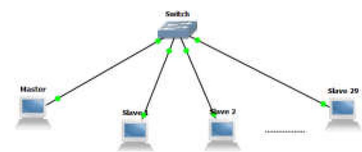


Fig. 3: Topology of laboratory setup

VI. Validation of Cooling Energy Integration

We validate out energy integration through comparing our simulation results against that obtained from real experiments performed in a test bed. We first briefly present the settings and then compare the results.

A. *Testbed Settings*

We prepare a testbed in a laboratory having thirty machines. Among them we select one as the master node and rest ones as slave nodes. Fig. 2 shows the distribution of tasks from the master nodes to slave nodes. To implement the distributed system, we have used Hadoop[18] framework. We have set up a distributed, multi-node(30 PCs) Apache Hadoop cluster backed by the Hadoop Distributed File System (HDFS), running on Ubuntu Linux[8]. We have varied the data size 3.14GB-12.6GB (with 4, 8, 12 and 16 files, each having 787 MB data) and the number of machines 5, 10, 15, 20, 15, 30 to get the total energy consumption and cooling energy consumption. For a particular data size and number of machines, we have run the popular word-count task four times so that we can get the desired values in average. Table I shows the lab environment in detail. Also, Fig. 4 shows the lab set-up. Also, from Fig. 3 we can see the overall topology where a master and 29 slaves are connected through a switch.

We use two Arduino energy monitors to get the computation energy and cooling energy. One monitor measures the CPU power and the other energy monitor measures the air-condition power. We use these hardware tools to estimate the values of real power, apparent power, power factor, voltage, current. Using some power formula we get the computation power and

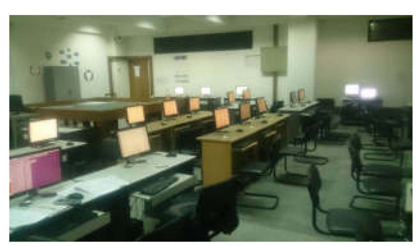


Fig. 4: Snapshot of laboratory setup

cooling power. Power consumed by air-conditions is considered as the cooling power. After that, using $Work = P \times T$ formula we get the computation energy and cooling energy where P refers to power and T refers to time. One thing is to be noted that, computation power means the power which is necessary to do the given task by the nodes. In the case of *SimGrid*, computation power can be measured by the existing energy plug-in. In this paper, we use computation power and energy consumption (without cooling energy) interchangeably.

B. Testbed Compatible Settings in SimGrid*:*

After setting up the lab testbed, we try to imitate the lab environment in our *SimGrid* environement. To imitate the testbed we need to make some conversion over units as *SimGrid* uses a different unit system for measuring machine power. *SimGrid* uses FLOating Points per Second (FLOPS) unit for evaluating the power of machines. We use the following formulation to evaluate FLOPS:
$FLOPS = N_s \times \frac{N_c}{N_s} \times C_e \times \frac{FLOPS}{C_y}$

Here, N_s refers to the number of chips, N_c refers to the number of cores, C_e refers to the number of clock cycle and C_y refers to the number of clock cycles. We adopt $N_s = 1$, $\frac{N_c}{N_s} = 2$, $\frac{FLOPS}{C_y} = 2$ in our simulation to be consistent with the laboratory setup. Table III shows the simulation environment.

From above formulation of $FLOPS$, we can see that if a machine has 2.4 GHz frequency then it will have $2.4 \times 2 \times 8 =$ 38.4 Giga FLOPS frequency, which can be written as 38400 Mega FLOPS. We use different number of files ranging from 4 to 16, each has 787 MB size. Hence, we consider data sizes in between 3.14GB - 12.6GB. In *SimGrid*, we have to give the number of chunks and the size of each chunk. We divide our total data size into five thousand chunks. For example, the 12.6GB data will be divided into equal sized five thousand chunks. This is a random value, however, other chunk sizes will give similar results. As we know from our early description that, each node can have three stages. From our experiment of testbed we know that most of the nodes consume 100W-120W power. The more the load, the more the power consumption. It is also known from our testbed experiment that when the machine is in an idle mode it will dissipate 40W of power. At power-off state, we have only 5W power consumption. In this experiment, we kept our data-size between 3.14 GB - 12.6 GB. We plan to run our experiments for big data files in future. *SimGrid* generally divides a large file into small chunks similar to Hadoop. Hence, if we use big files then the number of chunks will be increased, and the simulation will take much longer time to finish and will consume more cooling energy.

C. Experimental Results

We created an identical environment, and distributed same work among nodes. We did this process similarly in *SimGrid* and testbed experiment. After that we compare the cooling energy, computation energy, and total energy between testbed and *SimGrid*. We can see the comparison from Fig. 5-8 for various data size. For each data size these figures show the comparison for cooling energy, computation energy, and total energy. If we increase the number of machines the energy is decreasing, and this is evident in all the graphs. It is visible from the fact that all the graphs have a downward slope. If we use smaller number of machines, all of those machines need to work for a long time and thus increase the cooling energy. On the other hand, if we use more machines, then the jobs are distributed among all of the machines, the machines need to work for a smaller amount of time and decrease consumed energy.

TABLE III: Simulation environment in SimGrid

Parameter	Value
# of master machines	1
# of slave machines	4, 9, 14, 19, 24, 29
PC power	38,400-44,800 Mega FLOPS
PC power consumption	Peak: 100 - 120 W, idle: 40 W, power off: 5 W
Line bandwidth	100 kbps
Total # of files	4, 8, 12, 16
Size of each file	787 MB
Total data size	3.14. 6.3, 9.44, 12.6 GB
Maximum allowable temperature	22° C
Environment temperature	25° C

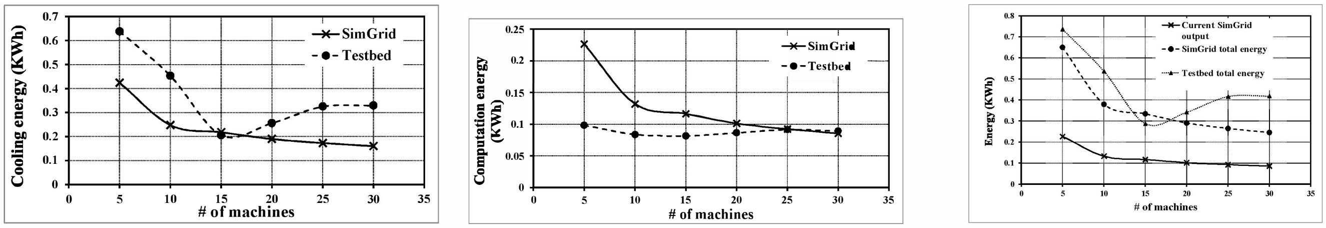
SimGrid previously had computation energy module. The comparison of the existing *SimGrid* module and the testbed can be seen from Fig. 5b. 6b, 7b, and 8b. We can see a common pattern between the line for *SimGrid* and testbed. The computation energy is decreasing as we increase the number of machines. Moreover, when the number of machines is 5, the difference between the computation energy value of *SimGrid* and testbed is bigger than the other differences. *SimGrid* and testbed have almost equal value when there are 15-20 machines.

Now we can see at the cooling energy comparison graphs from Fig. 5a. 6a, 7a, and 8a. We can see, all these cooling energy comparison graphs follow the similar pattern as the existing computation energy comparison graphs. These graphs also have a higher difference when the number of machines is 5, and they also show nearly same cooling energy when there are 15-20 machines. Hence, we can say that our modeled cooling energy curves follow the existing computation energy model curves.

We can see the total energy comparison from Fig. 5c. 6c, 7c, and 8c. Current *SimGrid* output line shows the total energy value if we do not integrate the cooling energy. *SimGrid* line shows the total energy after integrating cooling energy, and testbed total energy line shows the total energy for testbed. We can see from these graphs that if we do not integrate cooling energy, then the total energy value is far less than the actual value of the total energy. After integrating cooling energy we can get a compatible total energy curve between *SimGrid* and real testbed.

We can see from all the figures that both the results from *SimGrid* and testbed have some variations, however, we can get an idea of the computation and cooling energy based on the output from *SimGrid* and they are closely similar. If we do not integrate the cooling energy with *SimGrid*, this difference will be more bigger.

From Table II, we can see the comparison of average value and standard deviation between testbed and *SimGrid*. We compare the data for various workloads. We can see that average and standard deviation for cooling energy follows a similar pattern. Average value for computational energy is

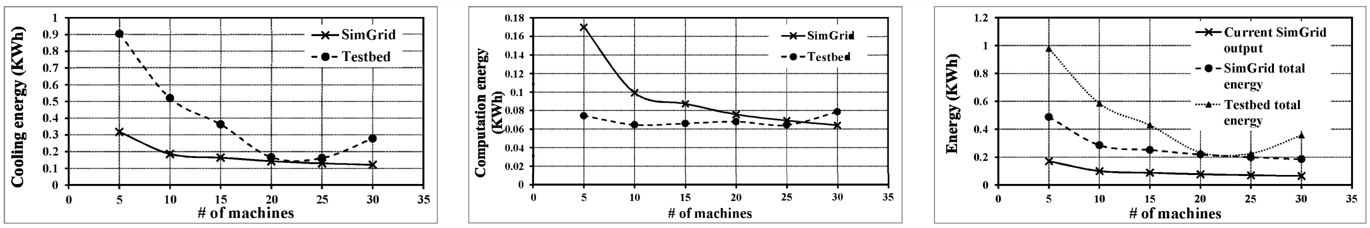


(a) Cooling energy comparison for 12.6 GB data

(b) Computation energy comparison for 12.6 GB data

(c) Total energy comparison for 12.6 GB data

Fig. 5: Energy comparison between *SimGrid* and Testbed for 12.6 GB data

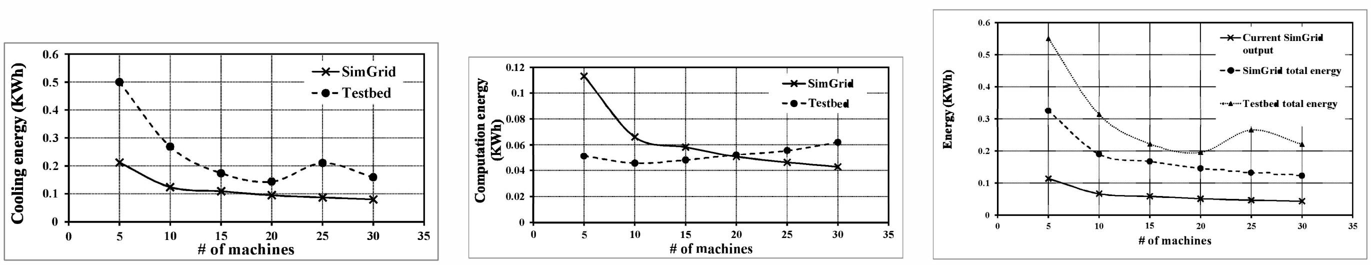


(a) Cooling energy comparison for 9.44 GB data

(b) Computation energy comparison for 9.44 GB data

(c) Total energy comparison for 9.44 GB data

Fig. 6: Energy comparison between *SimGrid* and Testbed for 9.44 GB data



(a) Cooling energy comparison for 6.3 GB data

(b) Computation energy comparison for 6.3 GB data

(c) Total energy comparison for 6.3 GB data

Fig. 7: Energy comparison between *SimGrid* and Testbed for 6.3 GB data

also is also similar. Variation in values for standard deviation in computational energy is very large. *SimGrid* computational energy consumption fluctuates more than real testbed data.

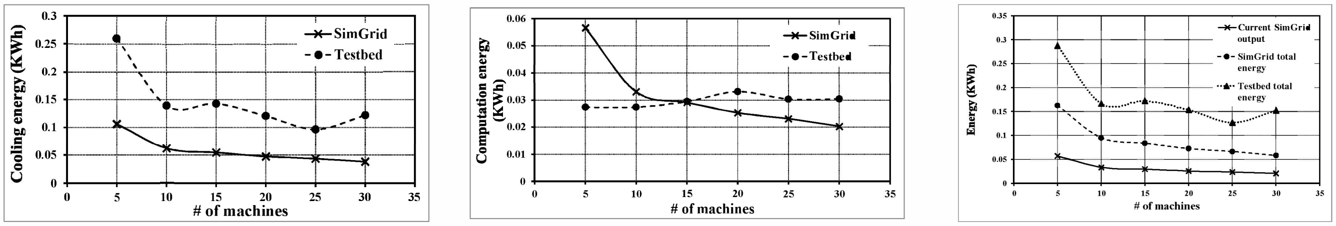
D. Summary of Findings

We can summarize our findings as follows:

- Both the computation energy and cooling energy deviate similarly from the testbed energy consumption. Hence, our module for cooling energy is compatible with the existing energy model of *SimGrid*.
- Number of machines and energy consumption follow an inverse relationship. This inverse relationship is observable in our cooling energy model as well as existing computation energy model. Testbed result also validates this.
- In most cases, if we increase the workload then the consumed energy is increased. We get the same trend for *SimGrid* simulation and testbed experiment, though there might be some exceptions.

VII. Conclusion

To run numerous distributed computing infrastructure such as data centers around the world, energy efficiency is highly desirable, as most such infrastructures use a huge amount of energy during their operation. To keep these infrastructures within a desirable level of temperature, significant amount of energy is consumed, which is known as the *cooling energy*. Hence, simulating tools for measuring cooling energy are necessary to imitate the real world environment of distributed systems. *SimGrid*, being a simulation tool for distributed systems, possesses an energy plug-in for measuring computational energy consumption by a distributed system.



(a) Cooling energy comparison for 3.14 GB data

(b) Computation energy comparison for 3.14 GB data

(c) Total energy comparison for 3.14 GB data

Fig. 8: Energy comparison between *SimGrid* and Testbed for 3.14 GB data

TABLE II: Average and standard deviation comparison between testbed and *SimGrid*

Workload	Cooling energy consumption(KWh)				Computational energy consumption(KWh)				% deviation in *SimGrid* w.r.t. testbed			
	Average		Std. deviation		Average		Std. deviation		Cooling energy		Computational energy	
	Testbed	*SimGrid*	Testbed	*SimGrid*	Testbed	*SimGrid*	Testbed	*SimGrid*	Average	Std. deviation	Average	Std. deviation
12.6 GB	0.3671	0.2347	0.1564	.0975	0.0883	0.125	0.006	0.052	36.07	37.7	41.6	766.67
9.44 GB	0.3977	0.176	0.2811	.0732	0.0693	0.094	0.006	0.039	55.7	74.0	35.6	550.0
6.3 GB	0.2419	0.117	0.1335	0.0488	0.0523	0.0627	0.0058	0.026	51.6	63.4	19.9	348.3
3.14 GB	0.1463	0.0583	0.0578	0.0247	0.0296	0.0311	0.0022	0.0132	60.15	48.58	5.07	500.0

However, this energy plug-in is yet to include any module for integrating cooling energy even though cooling energy plays a significant role within the total energy consumption. Therefore, in this paper, we attempt to integrate cooling energy consumption in *SimGrid*. Here, we present necessary models to be integrated and how to integrate them in *SimGrid*. Further, we simulate our integrated modules under different settings and compare the results with real testbed experimental results. Comparison between the simulation and real experimental results confirms applicability of our cooling energy integration in *SimGrid*.

References

[1] J.-C. Cano and P. Manzoni. A performance comparison of energy consumption for mobile ad hoc network routing protocols. In *Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000. Proceedings. 8th International Symposium on*, pages 57–64. IEEE, 2000.

[2] H. Casanova, A. Giersch, A. Legrand, M. Quinson, and F. Suter. Simgrid: a sustained effort for the versatile simulation of large scale distributed systems. *CoRR*, abs/1309.1630, 2013.

[3] J.-H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 22–31. IEEE, 2000.

[4] L. M. Feeney. An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. *Mobile Networks and Applications*, 6(3):239–249, 2001.

[5] S.-W. Ham, M.-H. Kim, B.-N. Choi, and J.-W. Jeong. Simplified server model to simulate data center cooling energy consumption. *Energy and Buildings*, 86:328–339, 2015.

[6] T. Hirofuchi and A. Lebre. Adding virtual machine abstractions into simgrid: a first step toward the simulation of infrastructure-as-a-service concerns. In *Cloud and Green Computing (CGC), 2013 Third International Conference on*, pages 175–180. IEEE, 2013.

[7] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen. A survey of energy efficient network protocols for wireless networks. *wireless networks*, 7(4):343–358, 2001.

[8] M. G. Noll. Running Hadoop on Ubuntu Linux. http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-multi-node-cluster/, 2016. [Online; accessed 22-February-2016].

[9] The Network Simulator NS-2. http://www.isi.edu/nsnam/ns/. [Online; accessed 15-August-2016].

[10] The Network Simulator NS-3. https://www.nsnam.org/. [Online; accessed 15-August-2016].

[11] A.-C. Orgerie, M. D. d. Assuncao, and L. Lefevre. A survey on techniques for improving the energy efficiency of large-scale distributed systems. *ACM Computing Surveys (CSUR)*, 46(4):47, 2014.

[12] E. Outin, J.-E. Dartois, O. Barais, and J.-L. Pazat. Enhancing cloud energy models for optimizing datacenters efficiency. In *Cloud and Autonomic Computing (ICCAC), 2015 International Conference on*, pages 93–100. IEEE, 2015.

[13] C. D. Patel, R. Sharma, C. E. Bash, and A. Beitelmal. Thermal considerations in cooling large scale high compute density data centers. In *Thermal and Thermomechanical Phenomena in Electronic Systems, 2002. ITHERM 2002. The Eighth Intersociety Conference on*, pages 767–776. IEEE, 2002.

[14] F. D. Rossi, M. G. Xavier, Y. J. Monti, and C. A. De Rose. On the impact of energy-efficient strategies in hpc clusters. In *2015 23rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, pages 17–21. IEEE, 2015.

[15] Sunon. How to Select the Right Fan or Blower. http://www.sunon.com/uFiles/file/03_products/07-Technology/004.pdf, 2016. [Online; accessed 15-August-2016].

[16] Traditionaloven.com. CFM to Watt Transformation. http://www.traditionaloven.com/tutorials/power/convert-atm-cfm-atmosphere-to-watts-w.html, 2016. [Online; accessed 22-February-2016].

[17] X. Wang, Y. Wang, and Y. Cui. A new multi-objective bi-level programming model for energy and locality aware multi-job scheduling in cloud computing. *Future Generation Comp. Syst.*, 36:91–101, 2014.

[18] T. White. *Hadoop: The Definitive Guide*. O'Reilly, first edition edition, june 2009.

[19] Q. Zhao, L. Tong, and D. Counsil. Energy-aware adaptive routing for large-scale ad hoc networks: Protocol and performance analysis. *Mobile Computing, IEEE Transactions on*, 6(9):1048–1059, 2007.

[20] Y. Zhu, M. Huang, S. Chen, and Y. Wang. Energy-efficient topology control in cooperative ad hoc networks. *Parallel and Distributed Systems, IEEE Transactions on*, 23(8):1480–1491, 2012.