Evaluating Impact of Live Migration on Data Center Energy Saving

Soramichi Akiyama*, Takahiro Hirofuchi†, Shinichi Honiden*‡
*Graduate School of Information Science and Technology, The University of Tokyo, Japan
Email: {akiyama, honiden}@nii.ac.jp
†National Institute of Advanced Industrial Science and Technology, Tsukuba, Ibaraki, Japan
Email: t.hirofuchi@aist.go.jp
‡National Institute of Informatics, Chiyoda, Tokyo, Japan

***Abstract*—Energy efficiency of cloud data centers is of great concern today and has been tackled by many researchers. Dynamic VM placement is a well-known strategy to improve energy efficiency of a data center. Virtual machines (VMs) under light load are consolidated into a small number of physical machines (PMs) to turn idle PMs into low-power states. Although live migration is essential for dynamic VM placement, former studies have not yet revealed how energy overhead of live migration has impact on energy efficiency of dynamic VM placement. To tackle this problem, we conducted integrated simulation of energy overhead of live migration and dynamic VM placement using SimGrid. We used three dynamic VM placement policies and two live migration mechanisms (existing pre-copy and an accelerated mechanism invented by us) to thoroughly evaluate the energy overhead. The results showed that in the worst case energy overhead of live migration occupies 5.8% of total energy consumption of a data center.**

***Keywords*-data center; live migration; energy efficiency;**

I. Introduction

A. Dynamic VM Placement to Save Data Center Energy

Cloud data centers are becoming more and more important as more and more people and enterprises use cloud services. In response to this trend, the amount of energy consumed by data centers has become enormous. Environmental Protection Agency of the U.S. reports that data centers consumed around 1.5 percent of total U.S. electricity consumption in 2006 [1].

Dynamic VM placement is a technique to dynamically switch execution hosts of VMs to improve efficiency of a data center, and reducing energy consumption of a data center is one of the important applications. Idle VMs are consolidated into small number of PMs to turn idle PMs into low-power states. Once the consolidated VMs become busy, they are distributed across many PMs to guarantee SLAs.

B. Energy Consumption of Live Migration

Live migration is a technique that allows a VM to move from one PM to another without interrupting services running on the VM. It is essential for dynamic VM placement because users of VMs never want their services to be stopped when VMs are replaced. However, live migration itself has energy consumption overhead. All the memory pages of the migrated VM are accessed and transferred through the network between the source and the destination PMs. This procedure increases load of CPU, memory, network and results in increased energy consumption.

Extra energy consumption of live migration has been discussed in [2]–[5]. Their findings include: (1) Both the source and the destination PMs undergo large increase of power as shown in [2] and [3]. (2) Not only CPU load but also memory and network load contributes to extra energy consumption because they occupy non-negligible parts of energy consumption within a server [6]. (3) Extra energy consumption is largely depends on the amount of transferred memory during migration as shown in [2] and [5].

C. The Problem and Approach

Evaluating energy saving achieved by a dynamic VM placement algorithm in a real data center requires not only analyzing the algorithm itself (energy reduction), but also considering extra energy consumption by live migration to execute it (energy overhead). They are studied separately by existing studies but integrated evaluations of them are missing. Some studies of dynamic VM placement consider the time and energy cost of live migration negligible [7], [8], and others take only the time cost of live migration into account [9], [10]. However, the energy overhead is not negligible as shown in this paper.

It is also important to study the same trade-off under the use of accelerated live migration mechanisms. There are enormous amount of researches on accelerated live migration mechanisms, and it is common in research to choose the best one depending on the characteristics. However, energy overhead of them and energy reduction achieved by integrating them and dynamic VM placement are not studied yet.

We tackle these two issues by following three steps. First, we integrate an energy model of pre-copy live migration into SimGrid [11]. We refer results from existing studies and exploit them to do this. Second, we build energy and performance models of an accelerated live migration mechanism, MiyakoDori [12], and integrate them into SimGrid. Finally, we use the modified SimGrid to conduct integrated evaluations of

This work is supported in part by Grant-in-Aid for JSPS Fellows and KAKENHI (25330097).

dynamic VM placement with extra energy consumption by live migration, and analyze the simulated results.

II. Modeling Performance and Energy Consumption of Live Migration

A. Performance and Energy Models

We create a performance model and an energy model of live migration to simulate dynamic VM placement with energy overhead of live migration taken into account.

Performance model estimates how long a live migration takes under a given environment. The input are the working set size of the target VM, network bandwidth available for migration, and workload running on the VM. This model is used to simulate dynamic VM placement algorithms to calculate how long each server can be turned into low-power states. *Energy model* estimates how much energy is lost by conducting live migrations to execute dynamic VM placement algorithms. In this particular study, the input is the amount of memory transferred during a migration (discussed later).

B. Performance and Energy Models of Pre-copy Migration

For normal pre-copy live migration, we use results from existing studies to build the models. Performance model of pre-copy live migration is discussed in [13] and is implemented in a well-known simulator SimGrid [11]. This model considers not only the allocated memory size of a migrated VM but also memory updates due to workloads running on the VM. Network resource contention is also considered.

Energy overhead incurred by pre-copy live migration is studied in [2]–[5]. Liu *et al.* [2] show that it increases in proportion to the amount of transferred memory during live migration. They also show that it does not depend on the available network bandwidth. The model is formulated as $E_{mig} = 0.512V_{mig} + 20.512$, where E_{mig} is the extra energy consumption in Joule, V_{mig} is the amount of transferred memory in the migration in Megabytes (equation (17) in [2]). The coefficients (0.512 and 20.512) changes depending on each physical servers. However, we confirmed that our servers consume similar amount of power with the ones used in [2] thus these values are used as is in this study.

C. MiyakoDori

MiyakoDori is an accelerated live migration mechanism developed by us [12]. MiyakoDori works effectively when a VM is migrated back to a PM where it has been executed before. This situation often happens in dynamic VM placement systems. Live migration with MiyakoDori works as follows:

1) When a VM V is migrated from PM_0 to PM_1, the memory image of V is kept undeleted in the memory space of PM_0 for future *reuse*.
2) While V is executed on PM_1, updates to V's memory pages are tracked using dirty page tracking.
3) When V is migrated back to PM_0, memory pages that has not been updated during step (2) are not transferred because PM_0 has a previous version of V's memory image. Other memory pages are normally transferred.

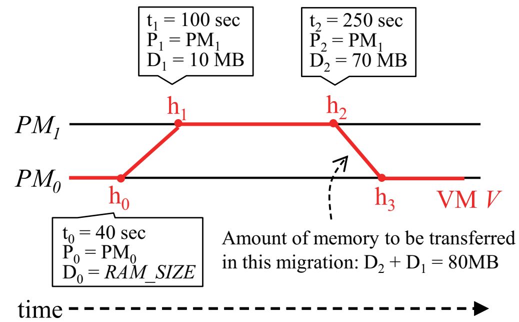

Fig. 1. Migration History.

Note that V's memory image is kept in the memory of PM_1 even after turning it to a low-power state, because rebooting a PM consumes much energy than turning it into sleep mode.

D. Performance Model of MiyakoDori

Developing a performance model of MiyakoDori requires a performance model of pre-copy live migration and calculating the amount of reusable memory in MiyakoDori. MiyakoDori can be simulated subtracting amount of reusable memory from the total memory usage of the migrated VM and then simulating pre-copy live migration, because it works totally the same as pre-copy live migration after reusing non-updated memory in the initial memory transfer. However emulating dirty page tracking to detect updated memory pages in SimGrid is infeasible.

We introduce migration history h_i of each VM into SimGrid to solve this issue. h_i includes a timestamp t_i, the VM's execution host P_i at t_i, and the amount of VM's updated memory D_i from t_{i-1} to t_i. A history is recorded every time a migration is started/finished. Figure 1 shows the overview of the migration history. Horizontal lines (PM_0, PM_1) show PMs that can host VMs. The bent line shows a VM migrated from PM_0 to PM_1, and then from PM_1 to PM_0. The amount of non-reusable memory in the 2$^{\text{nd}}$ migration is given by $D_2 + D_1$.

We confirmed that our performance model well simulates MiyakoDori by comparing total migration time when a VM is migrated back using the simulator and real implementation. The migrated VM executed a workload that dirties W MB of working set with D MB/s for T seconds. We tried {128, 256, 512, 1024} for W, {2, 4, 8} for D, and {10, 30, 60, 300} for T (48 combinations in total). The best fitting line between total migration time given by the simulator and the real implementation was $y = 0.95x+3.65$ (simulated values as x and real values as y). The coefficient of x is nearly 1, which means the model properly simulates the total migration time. The constant factor (+3.65) is because the real implementation takes 2–3 seconds for the preparation phase before starting actual memory transfer. Thus we add 2.5 seconds of waiting before starting a migration in the simulation.

E. Energy Model of MiyakoDori

We use the same energy model as normal pre-copy live migration for MiyakoDori. The overhead MiyakoDori incurs is

TABLE I
SIMULATION SETTINGS

Parameter	Value
Power of Active HPS (Watt)	$185 + (235 - 185) \times load/capacity$
Power of Sleep HPS (Watt)	20
Power of WHS (Watt)	$(235 + 185) \div 2$
# of HPSs / VMs	32 / 128 or 64 or 32
# of cores of a HPS / VM	4 / 1 or 2 or 4
NW bandwidth between PMs	10 Gbps
Memory size of a VM	4 GB

enough small to ignore in terms of energy consumption. Dirty page tracking is enabled even during non-migration time with MiyakoDori, but we confirmed it does not increase energy consumption by actual measurements using our servers. Extra data transfer for MiyakoDori is negligible compared to the amount of memory transferred in a migration. Note that although the model equation is the same, the amount of transferred memory (V_{mig}) does decrease with MiyakoDori thus the total energy consumption for a migration also decreases.

III. EXPERIMENTAL RESULTS

A. Metrics

For evaluation, we introduce three metrics. *Slept time ratio* is the average ratio of time during which a PM was in the low-power state against the whole duration of the simulation (12 hours). *Saved energy ratio* is the ratio of the amount of energy saved by dynamic VM consolidation. The values are calculated by subtracting actual energy consumption from expected energy consumption when all PMs are always on and live migration never occurs. *Energy overhead* is the ratio of energy consumption by live migration against the total energy used by all the servers in the data center. This value means the amount of wasted energy used for live migration.

B. Experimental Setup

Each VM on the simulated data center executes a bursty CPU-intensive workload, which has heavy and light CPU load phases alternating in short time periods. Specifically, the load of each VM changes as follows in our experiments:

1) Heavy load continues for a random period chosen from a range of 10 mins to 20 mins, with granularity of 100 seconds; i.e. 600 sec, 700 sec, ..., 1200 sec.
2) Then the load becomes light for a random period chosen from the same range.

Heavy and light load alternate within a simulation of 12 hours each. The memory of a VM is 4 GB and is updated with the speed of 2MB/s when the load is heavy.

VMs are dynamically placed across PMs using *Warehouse-highpower strategy*. It is a simple dynamic VM placement algorithm, where PMs are divided into two categories: *warehouse server (WHS)* and *high-power server (HPS)*. A WHS hosts idle VMs with overcommitment; i.e. a 4-core WHS hosts more than four 1-core VMs. A HPS hosts busy VMs with no overcommitment; i.e. a 4-core HPS hosts no more than four 1-core VMs. A VM is migrated from WHS to HPS when the load changes to heavy from light and the other way around when the load changes to light from heavy. The simulated data center has 1 WHS and 32 HPSs with 4 cores each.

How to select a target HPS depends on the operation policy of each data center, thus we try three policies. Note that this is to cover various operation policies, but not to show a specific one is better than others. *Most Dense* policy picks a HPS that is hosting the largest number of VMs but not exceeding the capacity. Reducing the total energy consumption is the primary concern in this policy. *Least Dense* policy picks a HPS that is hosting the smallest number of VMs. Reducing performance interference from other VMs is the primary concern in this policy. *Random* policy picks a HPS that is hosting the largest or the smallest number of VMs randomly. In this policy some VMs can be co-located without any interferences, while others must be scattered as much as possible.

The energy model of a HPS is built upon power measured with a real server with a 4-core Intel Xeon X5460, 8GB of memory, three 1 Gbps NICs, and an HDD. The server consumes 20 Watt, 185 Watt, and 235 Watt when it is in sleep mode, when it is on but idle, and when its 4 cores are fully loaded, respectively. Thus a simulated HPS is configured to consume $185+(235-185)\times load/capacity$ Watt when active, and 20 Watt when in sleep mode. The WHS is assumed to be moderately loaded and consume $(235 + 185) \div 2$ Watt.

C. Simulation Results and Analysis

Figure 2, Figure 3 and Figure 4 show Slept Time Ratio, Saved Energy Ratio and Energy Overhead, respectively. Bars indicated 32, 64, 128 VM are the results when each HPS can host 1, 2, or 4 VMs respectively because each VM has 4, 2, or 1 vCPUs. Experiment with mixed number of vCPUs is future work. All values are averaged across 30 simulations runs. Slept Time Ratio is calculated only across 32 HPSs. Saved Energy Ratio and Energy Overhead are calculated with the WHS taken into account.

1) Slept Time Ratio: In the 32VM cases, all policies show the same value because a PM can host only one VM thus the difference of the policy makes no change. However in the 128VM cases, Most Dense policy yields more than 10 times better results than Least Dense policy. This is because Least Dense policy distributes VMs as much as possible to prevent performance interference. Note that our intention is not to state Most Dense policy is better than Least Dense policy, but to evaluate the impact of live migration on various operational policies. MiyakoDori does not contribute much on Slept Time Ratio because the network bandwidth between PMs is large.

2) Saved Energy Ratio: The most important point is that saved energy ratio is greatly smaller than slept time ratio because of extra energy consumption by live migration as focused on this paper. In 128VM cases in Least Dense and Random, the values are negative even though PMs are in sleep mode for positive amount of time. It means in these cases always keeping all PMs active consumes smaller amount of energy than using dynamic VM consolidation. This is a good

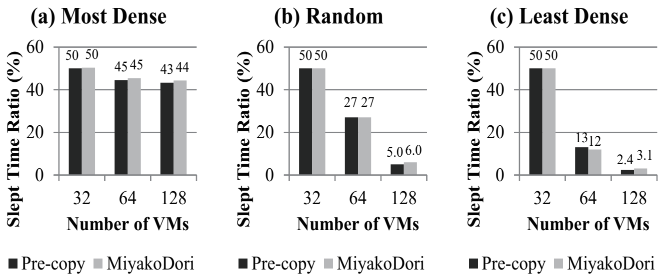

Fig. 2. Slept Time Ratio in Various VM Placement Policies.

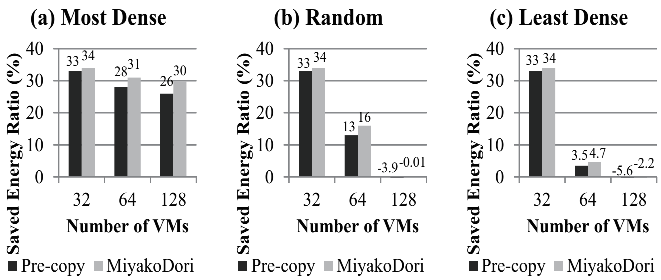

Fig. 3. Saved Energy Ratio in Various VM Placement Policies.

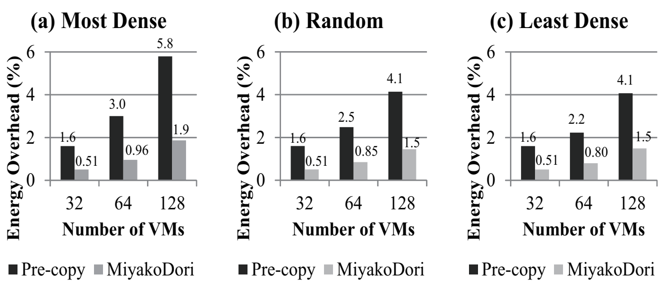

Fig. 4. Energy Overhead in Various VM Placement Policies.

example of our idea that impact of live migration on data center energy saving must be considered carefully.

3) Energy Overhead: The values show the ratio of wasted energy for live migration to the energy used for fruitful computation. Note that the values are not equal to the differences between slept time ratio and saved energy ratio because they include the amount of energy consumed by PMs in sleep mode and by the WHS. In the 128VM cases, energy overhead is more than 4% in all policies. This is not negligible at all because the overall energy consumption of data centers is huge, and IT equipment consumes 50% of the energy consumed in a traditional data center [6]. MiyakoDori decreases the energy overhead to less than 1.9%. These values give great insights to power and cost management of data centers.

IV. Related Work

Liu *et al.* [2] showed that extra energy consumption caused by the pre-copy live migration depends only on the amount of transferred. They gave detailed mathematical analysis but they never mention how the extra energy consumption impacts on the overall data center energy consumption. Aikema *et al.* [4] compared how extra energy consumption of live migration changes depending on workload type and transport type. They concluded live migration is "not always be advisable", but did not evaluate how much impact migrations have on the overall energy consumption. Hossain *et al.* [3] proposed a dynamic VM placement algorithm that considers extra energy consumption of live migration. However, they only showed the improvement by their new algorithm and do not analyze the energy overhead of live migration in detail. Goiri *et al.* [9] claimed cost of live migration must be considered as a penalty when conducting dynamic VM placement. They considered the time required to create a new VM and to migrate an existing VM when calculating energy-efficient VM placement, but does not care extra energy consumption of live migration.

V. Conclusion

We figured out existing researches of dynamic VM placement lack integrated evaluation of its energy reduction with energy overhead by live migration. We used performance and energy models of pre-copy live migration and an accelerated live migration mechanism, MiyakoDori, and simulated how they impact the overall energy consumption of a data center. We showed that the impacts are non-negligible and must be considered when energy saving of dynamic VM placement is discussed. Future work includes modeling other migration mechanisms and evaluating more sophisticated dynamic VM placement algorithms.

References

[1] "EPA Report on Server and Data Center Energy Efficiency." [Online]. Available: https://www.energystar.gov/index.cfm?c=prod_development.server_efficiency_study

[2] H. Liu, C.-Z. Xu, H. Jin, J. Gong, and X. Liao, "Performance and energy modeling for live migration of virtual machines," in *HPDC*, 2011.

[3] M. M. Hossain, J.-C. Huang, and H.-H. S. Lee, "Migration energy-aware workload consolidation in enterprise clouds," in *CloudCom*, 2012.

[4] D. Aikema, A. Mirtchovski, C. Kiddle, and R. Simmonds, "Green cloud vm migration: Power use analysis," in *International Green Computing Conference*, 2012.

[5] A. Strunk, "A lightweight model for estimating energy cost of live migration of virtual machines," in *IEEE International Conference on Cloud Computing*, 2013.

[6] L. A. Barroso, J. Clidaras, and U. Hölzle, *The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines*, 2nd ed. Morgan & Claypool Publisher, 2013.

[7] S. Srikantaiah, A. Kansal, and F. Zhao, "Energy aware consolidation for cloud computing," in *HotPower*, 2008.

[8] S. Lee, R. Panigrahy, and V. Prabhakaran, V. Ramasubramanian, "Validating heuristics for virtual machines consolidation," Microsoft Research, Tech. Rep. MSR-TR-2011-9, 2011.

[9] I. Goiri, F. Julià, R. Nou, J. Berral, J. Guitart, and J. Torres, "Energy-aware scheduling in virtualized datacenters," in *IEEE CLUSTER*, 2010.

[10] M. Y. Lim, F. Rawson, T. Bletsch, and V. W. Freeh, "Padd: Power aware domain distribution," in *ICDCS*, 2009.

[11] "SimGrid." [Online]. Available: http://simgrid.gforge.inria.fr/

[12] S. Akiyama, T. Hirofuchi, R. Takano, and S. Honiden, "Miyakodori: A memory reusing mechanism for dynamic vm consolidation," in *IEEE International Conference on Cloud Computing*, 2012.

[13] T. Hirofuchi, A. Lebre, and L. Pouilloux, "Adding a live migration model into simgrid: One more step toward the simulation of infrastructure-as-a-service concerns," in *CloudCom*, 2013.