Memory and Network Aware Scheduling of Virtual Machine Migrations*

Vincent Kherbache[†], Eric Madelaine[‡] Fabien Hermenier[§]

Live-migration has become a common operation on virtualized infrastructures. Indeed, it is widely used by resource management algorithms to distribute the load between servers and to reduce energy consumption [1, 2]. Operators rely also on migrations to prepare production servers for critical maintenance by relocating their running VMs elsewhere.

To apply new VM placement decisions, live-migrations must be scheduled by selecting for each migration the moment to start and the bandwidth to allocate. Long migrations violate SLAs and reduce the practical benefits of placement algorithms. The VMs should then be migrated as fast as possible. To do so, the migration scheduler must be able to predict accurately the migration durations and schedule them accordingly.

Dynamic VM placement algorithms focus extensively on computing a placement of quality. Their practical reactivity is however lowered by restrictive assumptions that under-estimate the migration durations [2]. For example, Entropy [1] supposes a non-blocking homogeneous network coupled with a null dirty page rate and we already demonstrated that the network topology but also the workload live memory usage are dominating factors [3]. Recently, some migration models have been developed and integrated into simulators to evaluate VM placement algorithms properly [4]. While these models reproduce migrations finely, they are only devoted to simulation purpose and not used to compute scheduling decisions.

We propose here a migration scheduler that considers the network topology, the migration routes, the VM memory usage and the dirty page rates, to compute precise migration durations and infer better schedules. We implemented our scheduler on top of BtrPlace [5], an extensible version of Entropy [1] that allows to enrich the scheduling decision capabilities through plug-ins. To assess the flexibility of our scheduler, we also implemented constraints to synchronize migrations, to establish precedence rules, to respect power budgets and an objective that minimizes energy consumption.

We evaluated our model accuracy and its resulting benefits by executing migration scenarios on a real testbed including a blocking network, mixed VM memory workloads and collocation settings. Our model predicted the migration durations with a 94% accuracy at minimum and an absolute error of 1 second while BtrPlace *vanilla* was only 30% accurate. This gain of precision led to wiser scheduling decisions. In practice, the migrations completed on average 3.5 time faster as compared to an execution based on BtrPlace *vanilla*. Thanks to a better control of migrations and power-switching actions we also reduced the power consumption of a server decommissioning scenario according to different power budgets.

*This work has been carried out within the European Project DC4Cities (FP7- ICT-2013.6.2).

[†]Phd. student

[‡]INRIA Sophia Antipolis, `firstname.lastname@inria.fr`

[§]Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, `fabien.hermenier@unice.fr`

References

[1] F. Hermenier, X. Lorca, J.-M. Menaud, G. Muller, and J. Lawall. Entropy: a Consolidation Manager for Clusters. In *VEE*, NY, USA, 2009. ACM.

[2] A. Verma, J. Bagrodia, and V. Jaiswal. Virtual Machine Consolidation in the Wild. In *Middleware'14*, New York, USA, 2014. ACM.

[3] V. Kherbache, E. Madelaine, and F. Hermenier. Planning Live-Migrations to Prepare Servers for Maintenance. In *Euro-Par: Parallel Processing Workshops*, volume 8806 of *LNCS*. Springer, 2014.

[4] T. Hirofuchi, A. Lèbre, and L. Pouilloux. Adding a Live Migration Model Into SimGrid, One More Step Toward the Simulation of Infrastructure-as-a-Service Concerns. In *IEEE CloudCom*, 2013.

[5] F. Hermenier, J. Lawall, and G. Muller. BtrPlace: A Flexible Consolidation Manager for Highly Available Applications. *IEEE Transactions on Dependable and Secure Computing*, 10(5), 2013.

Memory and Network Aware Scheduling of Virtual Machine Migrations

Vincent Kherbache
vincent.kherbache@inria.fr

Fabien Hermenier

Eric Madelaine

The benefits of live-migration scheduling

Live-migrations enact the efficient VM placements computed by load balancing and consolidation algorithms. They also allow to prepare production nodes for critical maintenance by relocating their running VMs elsewhere.

When several live-migrations must be executed, they should be scheduled by selecting for each migration, the best moment to start and the amount of bandwidth to allocate.

A good schedule will allow to migrate all the VMs as fast as possible by minimizing their effective duration and reducing the side effects due to the migration process itself.

Bad migration scheduling leads to SLA violations

Current migration models are too theoretical. They tend to over-simplify the complexity of the underlying infrastructure as well as the behavior of the live-migration algorithm.

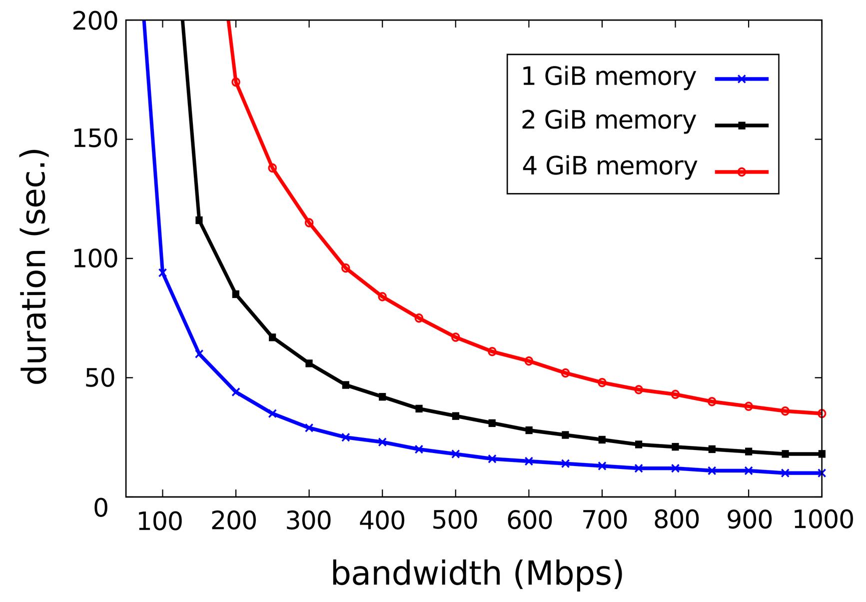
These simplifications lead to under-estimate the duration of such performance-costly operations, thereby resulting in SLA violations and uncontrolable migration actions. Thus making current placement algorithms less interesting than in theory.

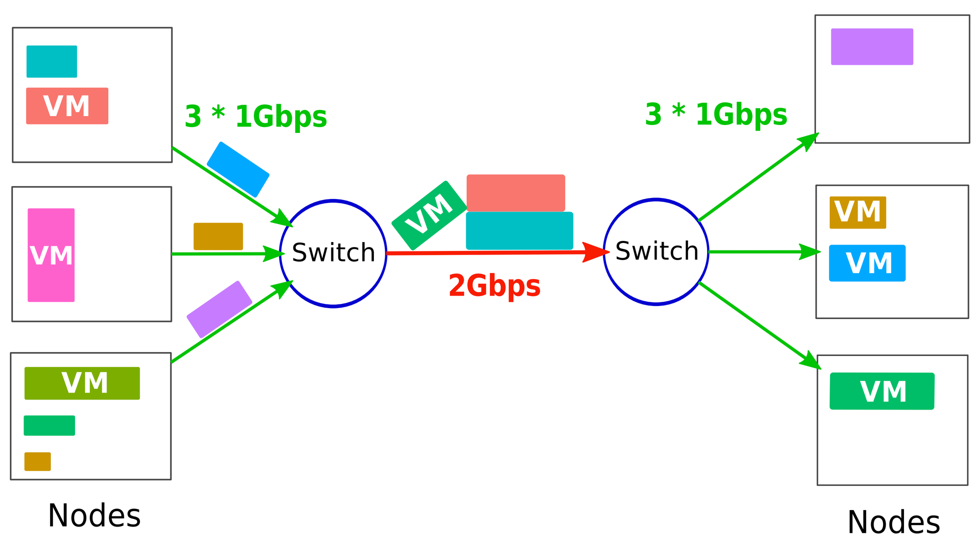
To improve the benefits of new VM placements but also to control the actions, *e.g.* to address energy awareness, we need to design finer migration models to take better decisions.

Our scheduler embeds a realistic migration and network model

Migrations are modeled

against the allocated bandwidth and live memory usage & using a blocking network and routing policies





Those models predict migration duration and ensure convergence under desired downtime

Implementation on top of BtrPlace

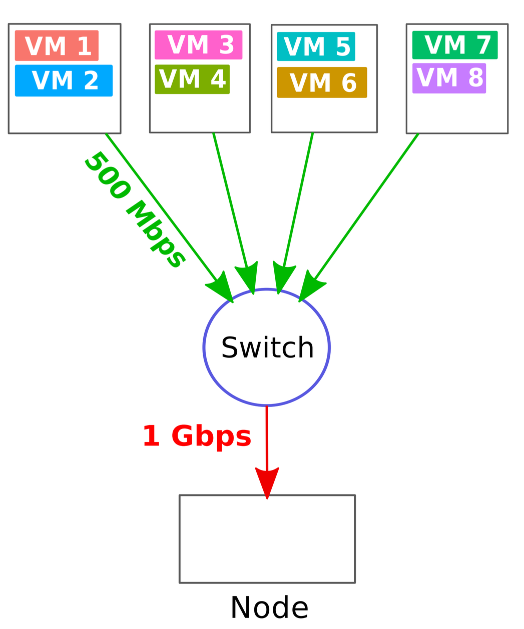
A VM manager where scheduling decisions can be enhanced through plugins using constraint programming

New adhoc constraints intended to tune the scheduling:

```
sync(vm[1-4]);
seq(vm[5,8]);
before(vm1,vm7);
powerBudget(500 Watts, [23:00-06:30]);
...
```

A wiser intra- and inter-node migration parallelism

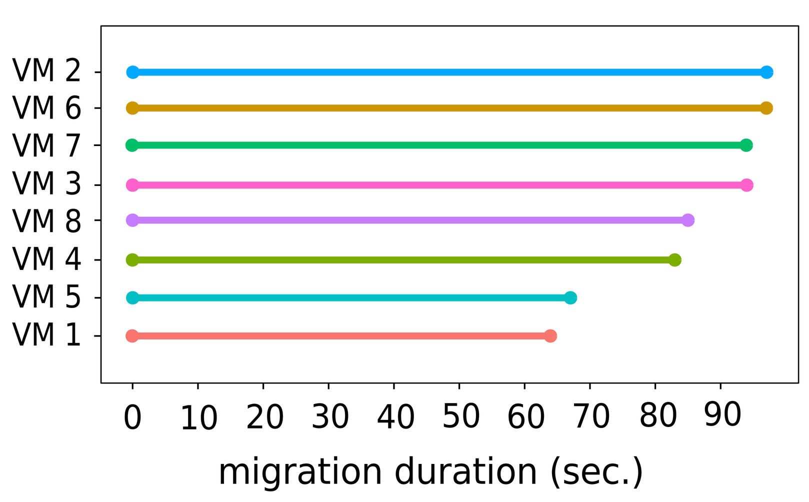
Experiment setup



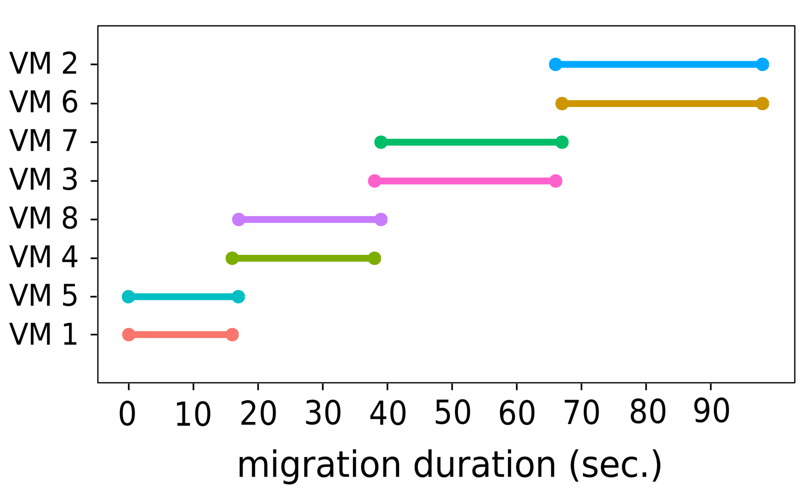
	Accuracy	Mean error
BtrPlace *vanilla*	**27.82 %**	60.9 s
BtrPlace + *scheduler*	**93.86 %**	1.5 s

Grid'5000

BtrPlace *vanilla*



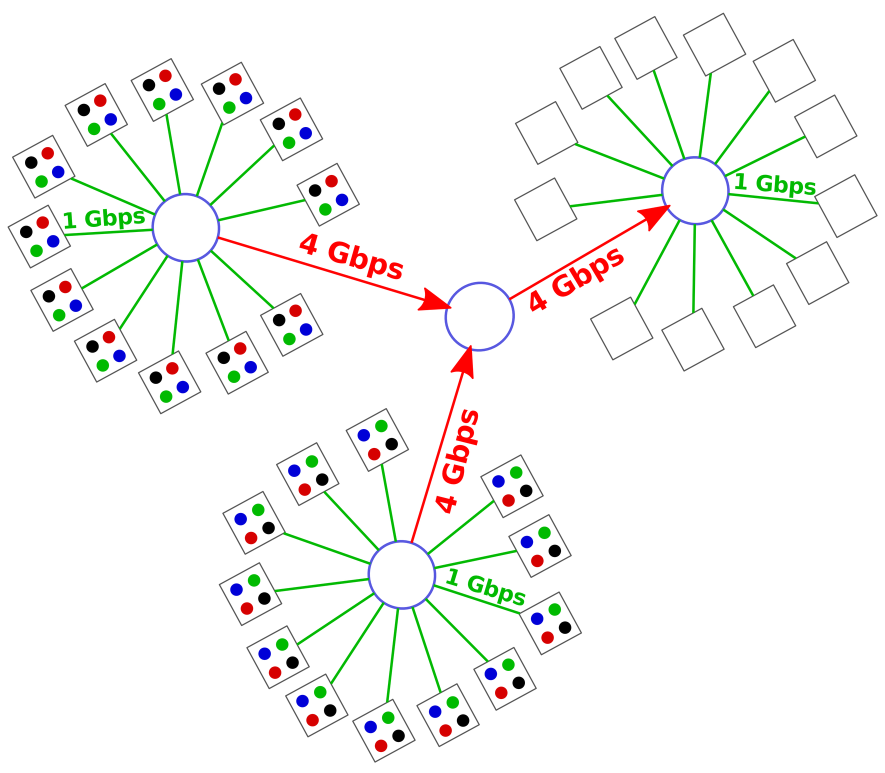
BtrPlace + *scheduler*



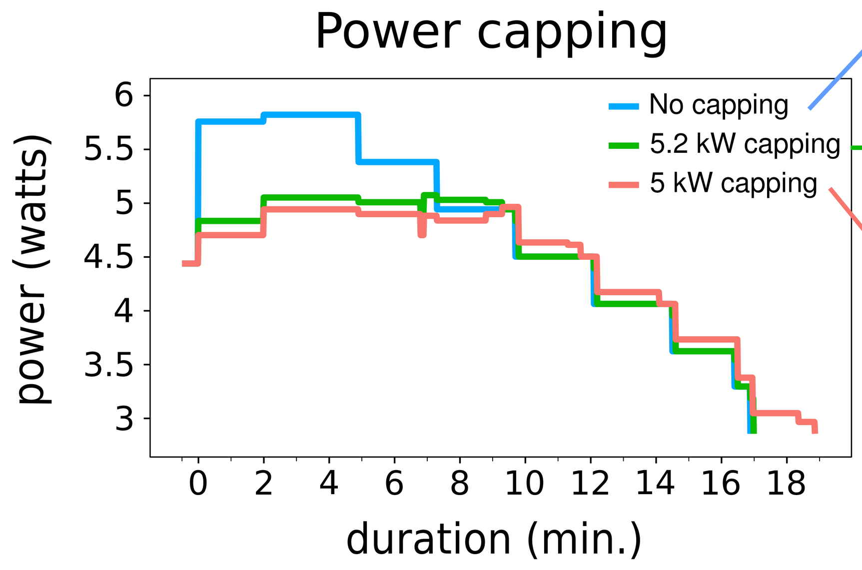
Migrations terminate 3.5 times faster

Use-case: Energy adaptive node decommissioning

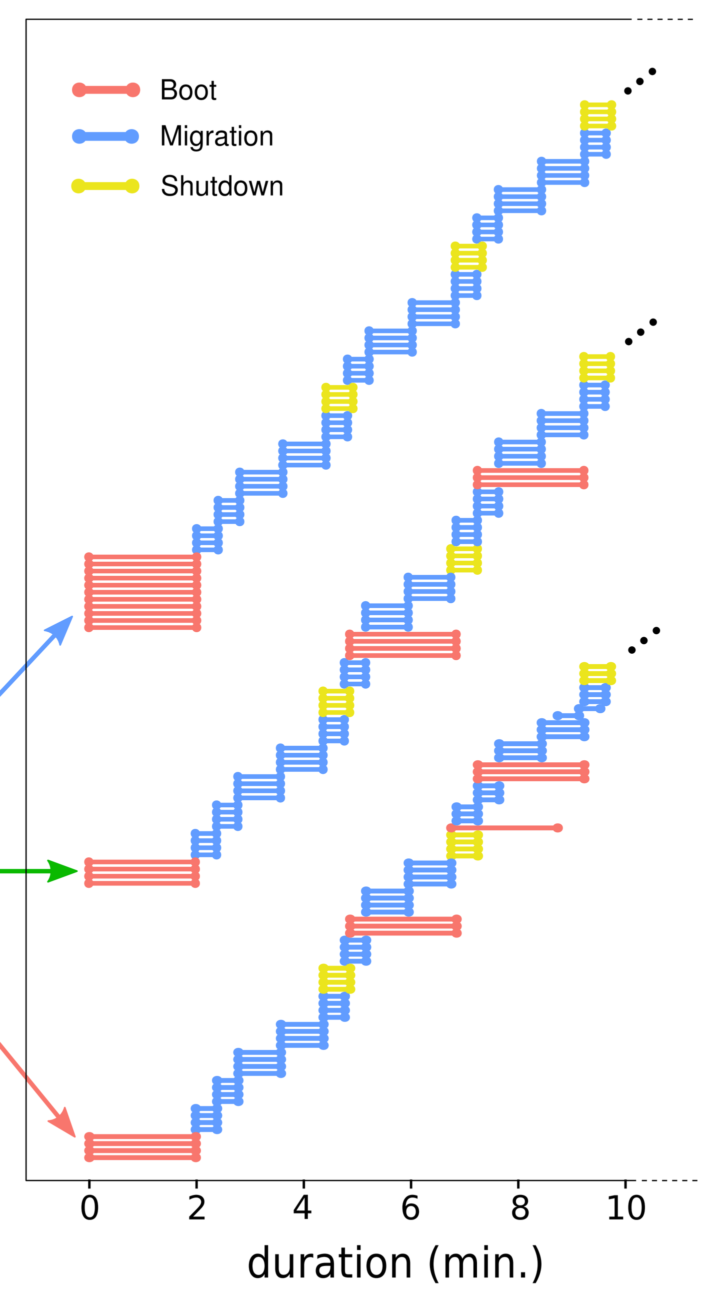
Migrations and power-switching actions are synchronized or delayed to fit a power budget



Power capping



Gantt charts



Inria — Université Nice Sophia Antipolis — DC4Cities