Indirect network impact on the energy consumption in multi-clouds for follow-the-renewables approaches

Miguel F. S. Vasconcelos[1,2] Daniel Cordeiro[1] Fanny Dufossé[2]

International Conference on Smart Cities and Green ICT Systems (**SMARTGREENS**), April 2022

University of São Paulo[1], Brazil
Université Grenoble Alpes[2], France

Table of Contents

Introduction

Cloud computing

- Essential component for our modern society
- Indirect network impact on the energy consumption in **multi-clouds** for follow-the-renewables approaches
    - Geographic distribution of data centers (DCs)

Figure 1: Locations of Microsoft Azure DCs.

The problem

- DCs consume $\approx$ 1% of the global power ($\approx$ 200 TWh in 2020)
- Most energy comes from nonrenewable (brown) sources:
    - *Data Center Alley* handled 70% of the internet data traffic (2019)
    - Supplied by 2% of renewable energy

Reducing the environmental impact of DCs operation

- Adopting renewable (green) energy into DCs
  - Projects already deployed (or in development) by major cloud providers (Amazon AWS, Apple, Facebook, Google, Microsoft)
- **Intermittent nature of green sources**
  - Solar power production only during the day
  - Wind power production only when the wind is blowing

Follow-the-renewables

Indirect network impact on the energy consumption in multi-clouds for follow-the-renewables approaches

- Virtualization and live-migrations
- Allocates/Migrates the workload to the DCs that have more renewable (green) power available

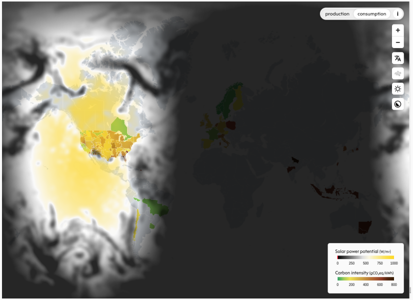


Figure 2: electricityMap - Solar irradiation.

Network usage by live-migrations

Indirect network impact on the energy consumption in multi-clouds for follow-the-renewables approaches

- Energy consumption of network devices can be considered constant (direct impact)
- Migrating the workload among different DCs generates extra computations proportional to the duration of the migration (indirect impact)

Network usage by live-migrations

Indirect network impact on the energy consumption in multi-clouds for follow-the-renewables approaches

- Energy consumption of network devices can be considered constant (direct impact)
- Migrating the workload among different DCs generates extra computations proportional to the duration of the migration **(indirect impact)**
    - **network congestion**
    - **energy consumption**

NEMESIS and c-NEMESIS

"Network-aware Energy-efficient Management framework for distributEd cloudS Infrastructures with on-Site photovoltaic production"[1]

- Resource management framework with a central controller
- Stochastic green and brown power consumption prediction
- Greedy heuristics for the scheduling
- Follow-the-renewables for workload allocation and migration
- Servers consolidation
- **Basis of this work**

[1] B. Camus et al. "Network-Aware Energy-Efficient Virtual Machine Management in Distributed Cloud Infrastructures with On-Site Photovoltaic Production". In: *2018 30th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. Lyon, France: IEEE, 2018, pp. 86–92.

NEMESIS

- Algorithm has 4 steps:
  - **Pre-allocation of incoming Virtual Machines (VMs)**: tries to allocate the VM into the server that is expected to consume the least brown energy

NEMESIS

- Algorithm has 4 steps:
  - **Pre-allocation of incoming Virtual Machines (VMs)**
  - **Revision of pre-allocations**: reviews the allocations given that greedy heuristics are fast, however may not provide the best solution

NEMESIS

- Algorithm has 4 steps:
  - **Pre-allocation of incoming Virtual Machines (VMs)**
  - **Revision of pre-allocations**
  - **Migration of the running VMs**: migrates the VMs from the brownest DCs to the greenest ones (inter-DC migrations). Restrictions considered: (i) the migration needs to finish during the considered time slot (5min); (ii) the VM will not finish its execution during the migration; (iii) one DC can only migrate to another 2 DCs during a time slot; and (iv) migrations from one DC are planned to execute one after another (no migrations in parallel)

NEMESIS

- Algorithm has 4 steps:
  - **Pre-allocation of incoming Virtual Machines (VMs)**
  - **Revision of pre-allocations**
  - **Migration of the running VMs**
  - **Servers consolidation**: performs live-migrations (intra-DC) to redistribute the workload among the servers trying to minimize the number of servers that are ON

c-NEMESIS

“Congestion and Network-aware Energy-efficient Management framework for distributEd cloudS Infrastructures with on-Site photovoltaic production”

- Algorithm has 4 steps:
    - **Pre-allocation of incoming Virtual Machines (VMs)**
    - **Revision of pre-allocations**
    - **Migration of the running VMs**
    - **Servers consolidation**

c-NEMESIS

- Modifications:
  - the bandwidth of the links, and the history of its usage is considered for the scheduling
  - inter-DC migrations are performed in parallel between DCs
  - intra-DC migrations do not execute simultaneously and are distributed in time (for each DC)
  - the estimation algorithm for the duration of migrations considers the real number of links that interconnects the origin and the target server

Simulations

Framework

- Simgrid (3.28)
  - Well-validated by the scientific community (over 20 years of usage)
  - Servers' power consumption uses a linear model based on CPU usage
  - Flow-level TCP modeling of the network
- Modification for modeling live-migration power consumption:
  - one CPU core is used in the target host during the VM migration process

Cloud platform

- Based on a real example: Grid'5000
- 1035 homogeneous servers distributed among 9 DCs
  - 2 x Intel Xeon E5-2630 (6 CPU cores per processor)
  - 32 GB RAM
- Network:
  - 1Gbps links intra DC
  - 10Gbps links inter DC

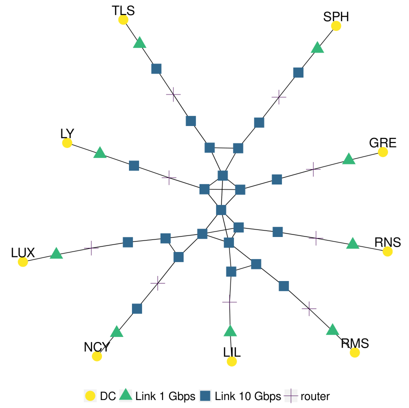


Figure 3: DCs and how they are connected in the network.

Green energy traces

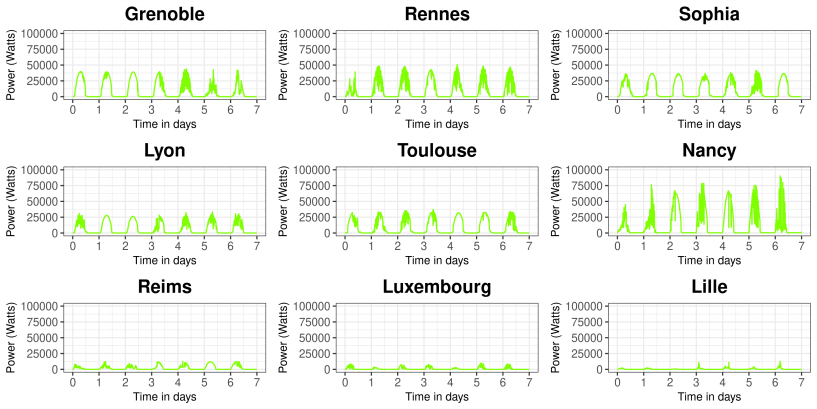


Figure 4: Green energy power production per DC - Source of data: Photovolta projec.

Workloads

- Virtual machines
- Traces samples from real cloud providers:
    - Google (2011): 380k VMs
    - Azure (2020): 300k VMs
- Information extracted:
    - Submission time, CPU cores requested, runtime
- RAM = 2GB per CPU cores (t2.small)
- No network usage

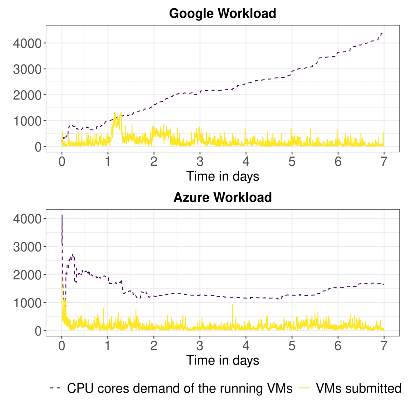


Figure 5: Workloads used for the simulations.

Results

Results

- Analysis of the live-migrations
  - Network congestion and wasted energy
- Total and brown energy consumption
- Comparison with two other state-of-the-art works

Baselines

WSNB (**W**orkload **s**hifting **n**on **b**rownout)[2]:

- Allocates the workload to the nearest DC that has available green power
- Follow-the-renewables strategy applied for the initial allocation
- Does not perform live-migrations
- Does not shutdown under-utilized servers

[2] Minxian Xu and Rajkumar Buyya. "Managing renewable energy and carbon footprint in multi-cloud computing environments". In: *Journal of Parallel and Distributed Computing* 135 (2020), pp. 191–202. ISSN: 0743-7315.

Baselines

FollowME@Source[3]:

- Allocation step: tries to allocate the incoming VMs to the greenest DC
- Migration step: Either only intra (origin = destination) or inter (origin != destination) DC
  - Intra DC: executed at each DC separately
  - Inter DC: tries to migrate the workload to the greenest DC
- Under-utilized servers are shut down (server consolidation)
- Do not consider network for migration planning

[3] Hashim Ali et al. "FollowMe@LS: Electricity price and source aware resource management in geographically distributed heterogeneous datacenters". In: *Journal of Systems and Software* 175 (2021), p. 110907. ISSN: 0164-1212.

Baselines

Follow-the-renewables strategy

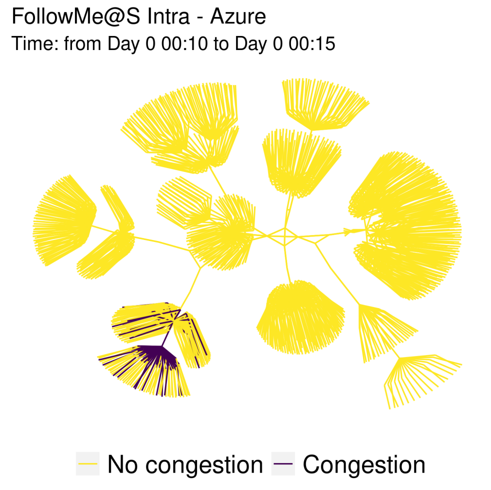
- Only for the VM allocation
  - WSNB and FollowME@S Intra
- During the whole execution of the workload
  - NEMESIS, c-NEMESIS and FollowME@S Inter

Assessing network congestion

- Comparison with a **"perfect scenario"**:
  - All migrations are executed again individually
  - Full access to network resources
- Additional time the migration takes (when planned by the scheduling algorithms) in comparison with the perfect scenario

Visualizating network congestion

- A link would be under congestion if the migration took more than 10% compared to the "perfect scenario"



Visualizating network congestion

Comparison of the different algorithms

Network congestion

Table 1: Extra seconds during migrations compared to the case when there is no congestion for the Azure workload, where "avg." stands for the average of the observations, "max." for the maximum value, and "rel." for the relative value.

Algorithm	**avg. rel.**	**max. rel.**	**Total extra seconds**
NEMESIS	1.6	3.98	86235.5
c-NEMESIS	1.0	1.32	4224.4
FollowME@S Intra	4.4	25.56	16384188.8
FollowME@S Inter	7.8	157.24	18531893.3

Wasted energy

Table 2: Wasted energy in the migrations (Wh) for the Azure workload.

Algorithm	**Origin**	**Target**
NEMESIS	539.6	491.1
c-NEMESIS	39.3	24.1
FollowMe@S Intra	163128.1	93298.9
FollowMe@S Inter	175086.3	105528.8

Wasted energy

- Wasted energy in the migrations could power the cloud infrastructure:
- 367 kWh of green energy was wasted in the case of the FollowMe@S Intra algorithm with the Google workload
- This energy could have powered the Luxembourg DC (38 servers at maximum capacity) for approximately 44 hours

Total and brown energy consumption

Table 3: Comparison of energy consumption (MWh) for the Azure workload.

Algorithm	**Total**	**Brown**
NEMESIS	30.43	21.21
c-NEMESIS	30.55	21.20
FollowMe@S Intra	31.69	22.41
FollowMe@S Inter	31.69	22.40
WSNB	33.56	24.23

Conclusions and Future work

Conclusions and Future work

- **Conclusions**
  - Bad migration planning results in network congestion, waste of green energy (and increase in brown energy consumption)
  - Follow-the-renewables approaches need to consider all the workload execution, given the intermittent nature of renewables
- **Future Work**
  - Network usage by the workload
  - Other virtualization techniques (containers)
  - Shutting down network equipment vs. network congestion

Thank you!

Indirect network impact on the energy consumption in multi-clouds for follow-the-renewables approaches

Contact: miguel.silva-vasconcelos@inria.fr

Figure 7: Repository with presentation, paper and experiments!

Wasted energy

$$wastedOrigin = 0 \ wastedTarget = 0 \ \forall migration \in migrations$$
$$extraTime = migration.time - migration.perfect$$
$$wastedOrigin+ = extraTime * powerPerCore * vmCores$$
$$wastedTarget+ = extraTime * powerPerCore$$

Accuracy of the estimation algorithm

Metrics: Mean Absolute Percentage Error (M.A.P.E.):
$\frac{1}{n}\sum_{i=1}^{n}\frac{|R_i - F_i|}{R_i}$

- a percentage value, and it represents the relative value of the estimation errors compared to the original value

Root Mean Square Error (R.M.S.E.): $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(R_i - F_i)^2}$

- metric similar to the standard deviation, and it allows to validate how far from the original value was the estimation

where: n represents the amount of values being considered, i the index of the value being considered, R_i the real duration of migration, and F_i the estimated duration

References

Ali, Hashim et al. "FollowMe@LS: Electricity price and source aware resource management in geographically distributed heterogeneous datacenters". In: *Journal of Systems and Software* 175 (2021), p. 110907. ISSN: 0164-1212.

Camus, B. et al. "Network-Aware Energy-Efficient Virtual Machine Management in Distributed Cloud Infrastructures with On-Site Photovoltaic Production". In: *2018 30th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. Lyon, France: IEEE, 2018, pp. 86–92.

Xu, Minxian and Rajkumar Buyya. "Managing renewable energy and carbon footprint in multi-cloud computing environments". In: *Journal of Parallel and Distributed Computing* 135 (2020), pp. 191–202. ISSN: 0743-7315.