Hierarchical Approach for Green Workload Management in Distributed Data Centers

Agostino Forestiero, Carlo Mastroianni, Giuseppe Papuzzo, Mehdi Sheikhalishahi

Institute for High Performance Computing and Networks, Italy

Spin-off from Italian CNR
http://www.eco4cloud.com

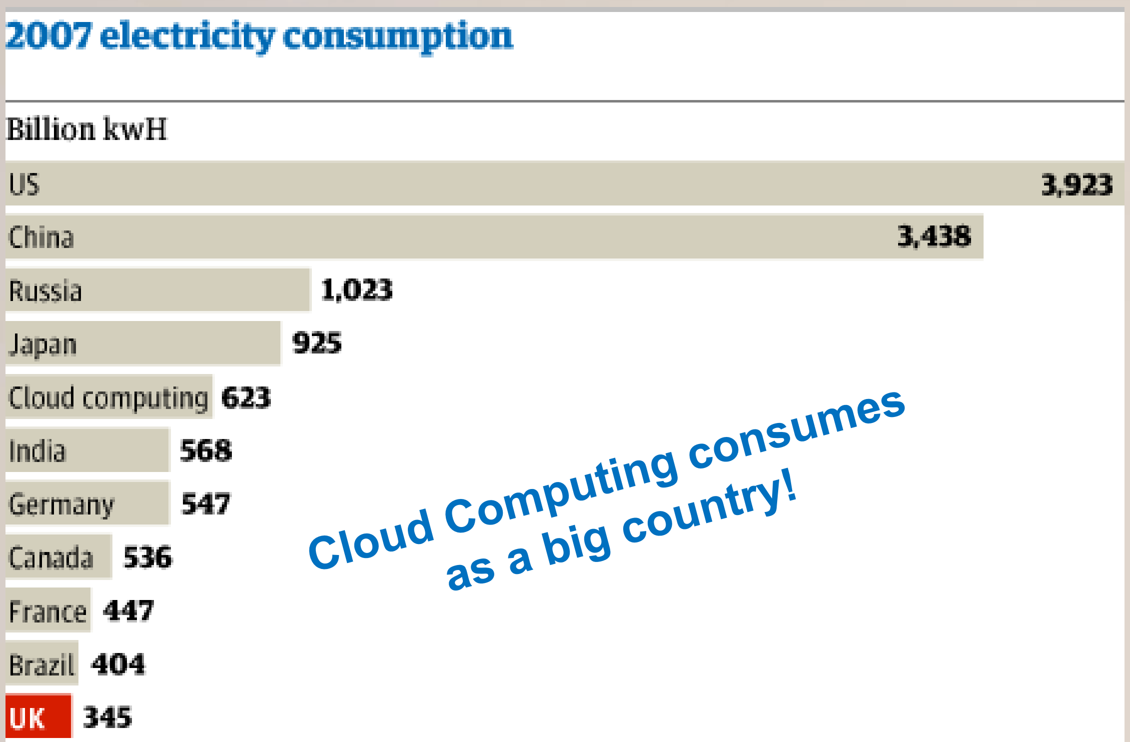
Michela Meo

Politecnico di Torino, Italy

The Energy Problem: contribution of ICT

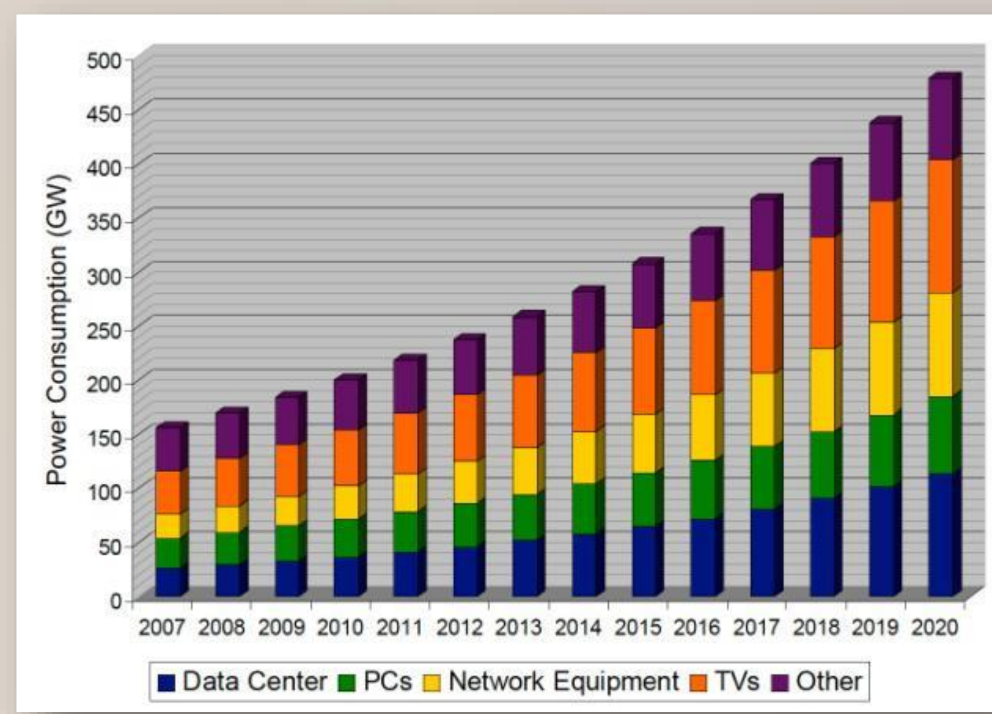
The ICT sector:

- accounts for **~3%** of total energy consumption worldwide, and is expected to double **every 5 years**
- produces between 2% and 3% of total emissions of greenhouse gases



Cloud Computing consumes as a big country!

***Source**: Greenpeace Report "How Clean is Your Cloud?", April 2012*



***Source**: Pickavet et al (IBBT, 2011)*

***Sources**: Pickavet & all (UGent-IBBT)*

Contribution of data centers is increasing

	Emissions 2007 ($MtCO_2e$)	Percentage 2007	Emissions 2020 ($MtCO_2e$)	Percentage 2020
World	830	100%	1430	100%
Server farms/Data Centres	116	$\boxed{14\%}$	257	$\boxed{18\%}$
Telecoms Infrastructure and devices	307	37%	358	25%
PCs and peripherals	407	49%	815	57%

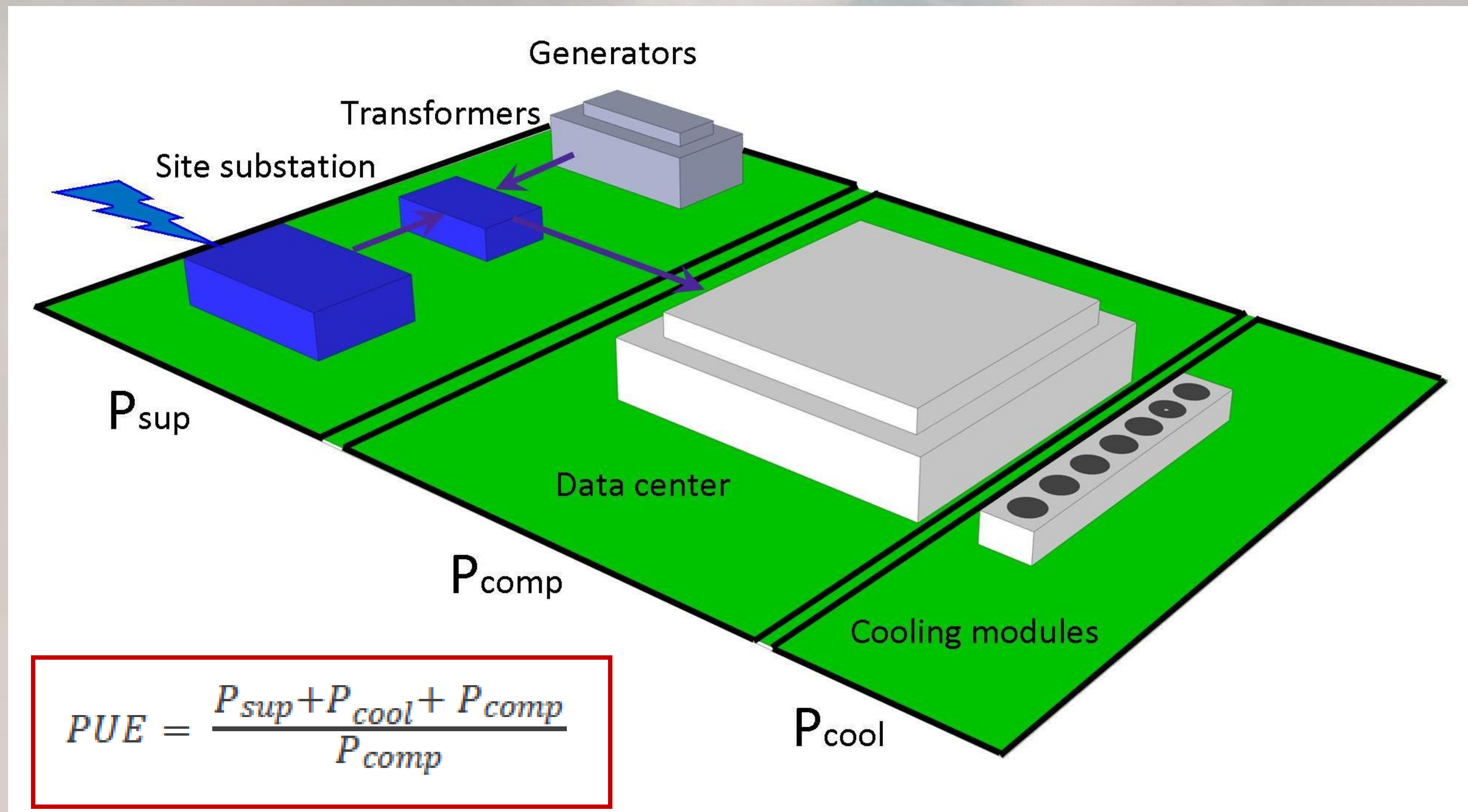
$MtCO_2e$ = Metric Tonne Carbon Dioxide Equivalent

Source: "Smart 2020: Enabling the Low-Carbon Economy in the Information Age", The Climate Group, June 2008.

Energy/cost savings opportunities

1. Improve infrastructure
   - use liquid cooling, improve efficiency of chillers and power supplies
2. Adopt more energy-efficient servers
   - feasible for CPU (e.g., using DVFS), on-going efforts on more efficient network utilization, little to do for RAM, disks, etc.
3. Consolidate VMs on fewer servers
   - unneeded servers can be hibernated or used to accommodate more load
   - consolidation should follow workload fluctuations (daily, weekly)
4. Move workload where energy is cheaper
   - “follow the moon”: feasible in distributed data centers

Improve physical infrastructure



Improving power distribution and cooling improves PUE, but has no impact on amount of power used for computation

Inefficient utilization of servers

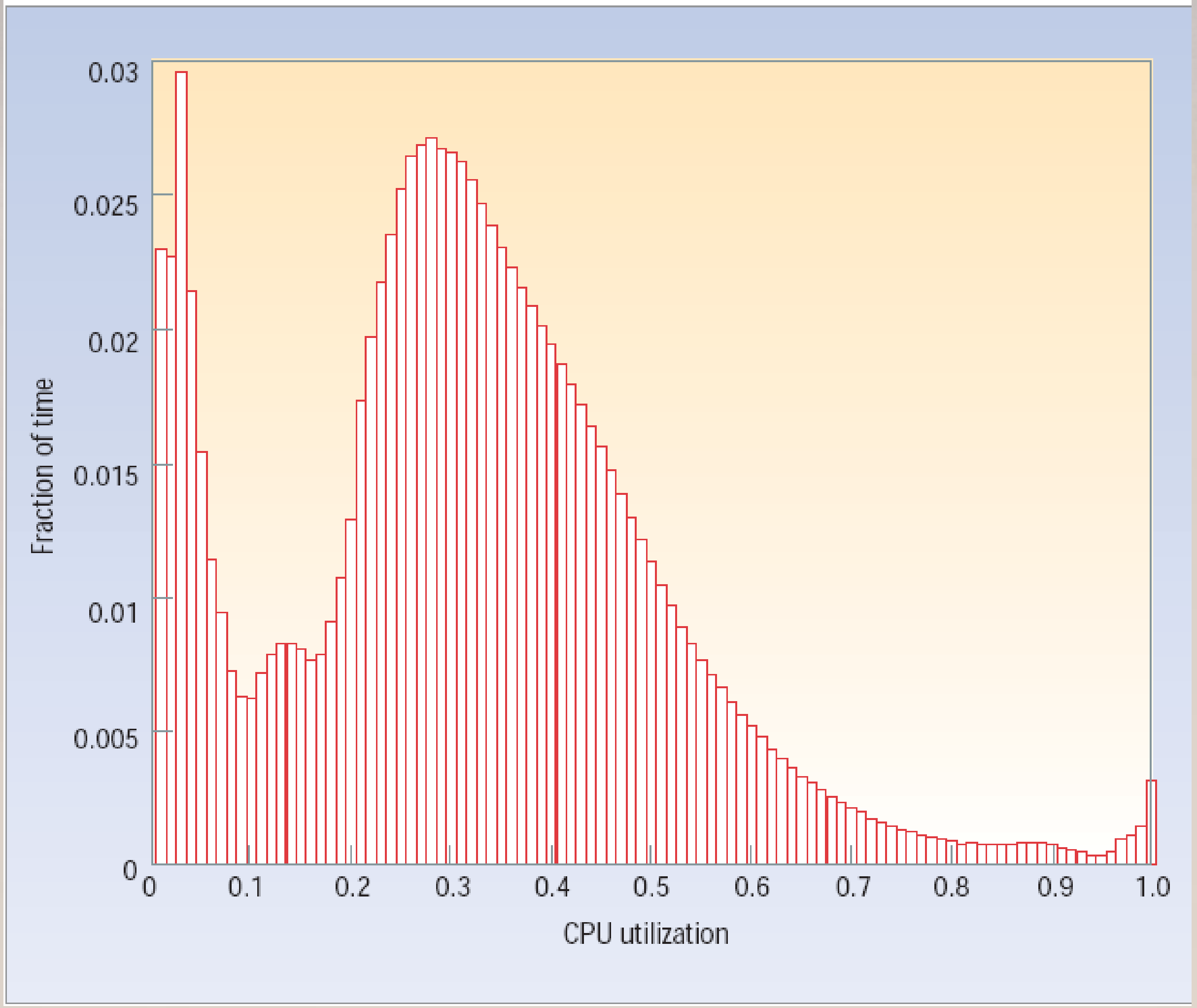
Two sources of inefficiency:

1) Servers are underutilized (between 15% and 30%)

2) An idle server consumes more than 50% of the energy consumed when fully utilized

This means that it is generally possible to consolidate the load on fewer and better utilized servers!

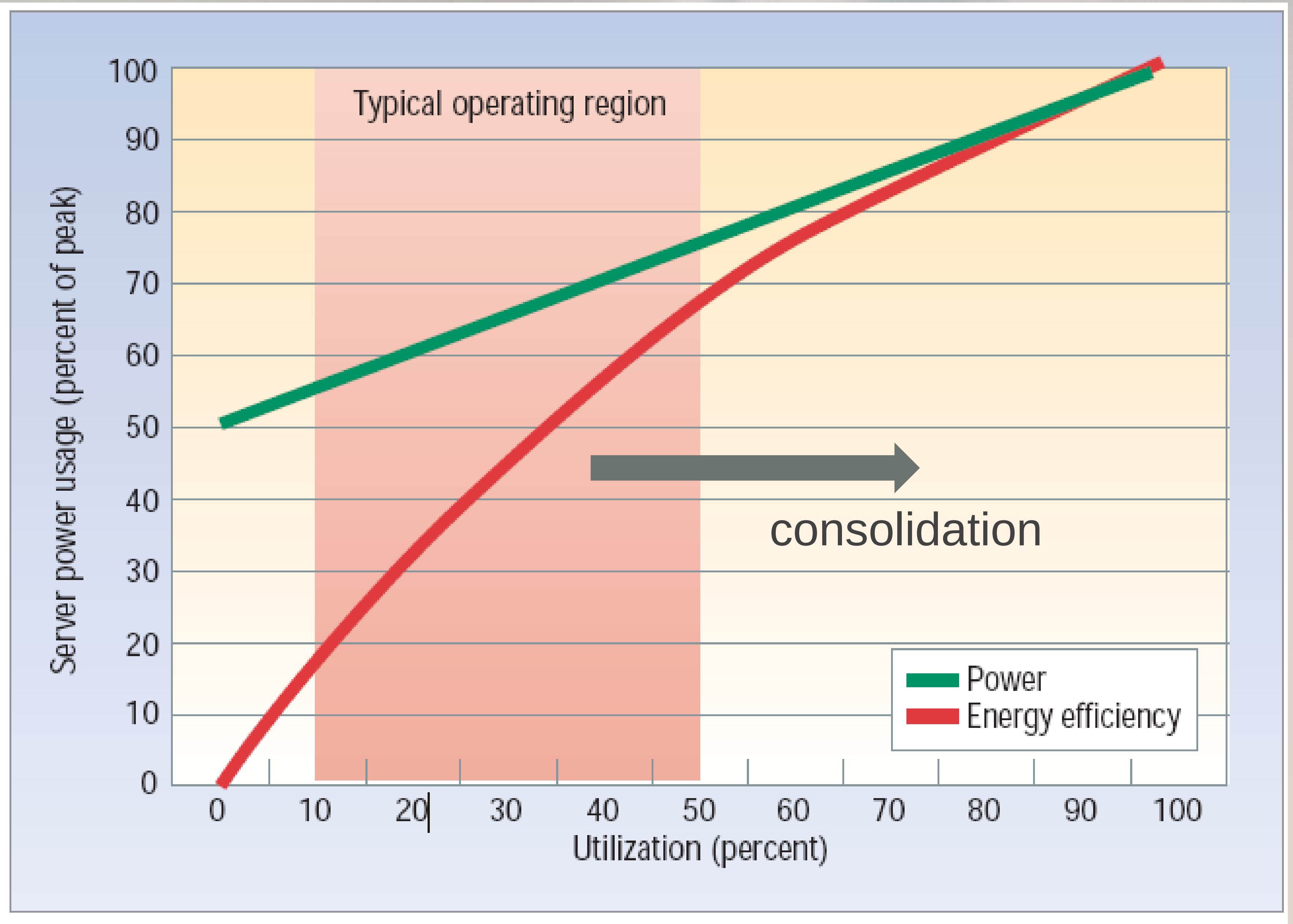
Typical utilization of servers



most servers are in 10% to 50% region of CPU utilization

Source: L.Barroso, U.Holzle, The case of energy proportional computing, ACM Computer Journal, Volume 40 Issue 12, December 2007.

Typical energy efficiency behavior



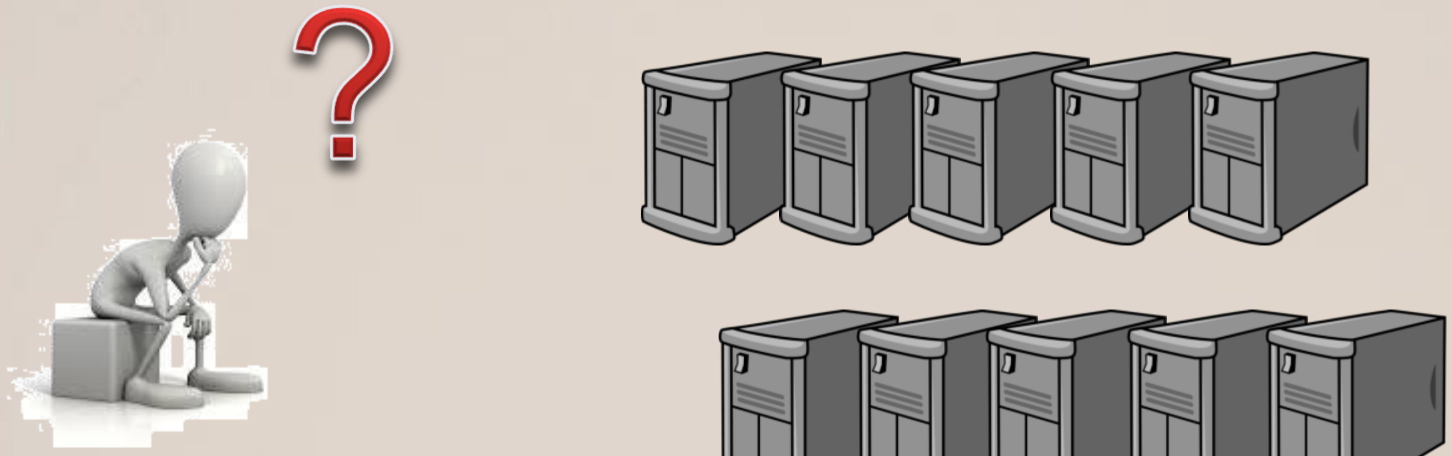
Energy efficiency is utilization divided by power consumption (useful workload/W)

Energy efficiency is low in the typical operating region

By consolidating the workload, the operating region is shifted to the right

Consolidation of VMs

- Assign VMs on the smallest number of servers
- NP-hard problem (online, multi-dimensional bin packing problem)
- Solutions available today are often complex, not scalable and may require a massive reassignment of VMs



Eco4Cloud solution

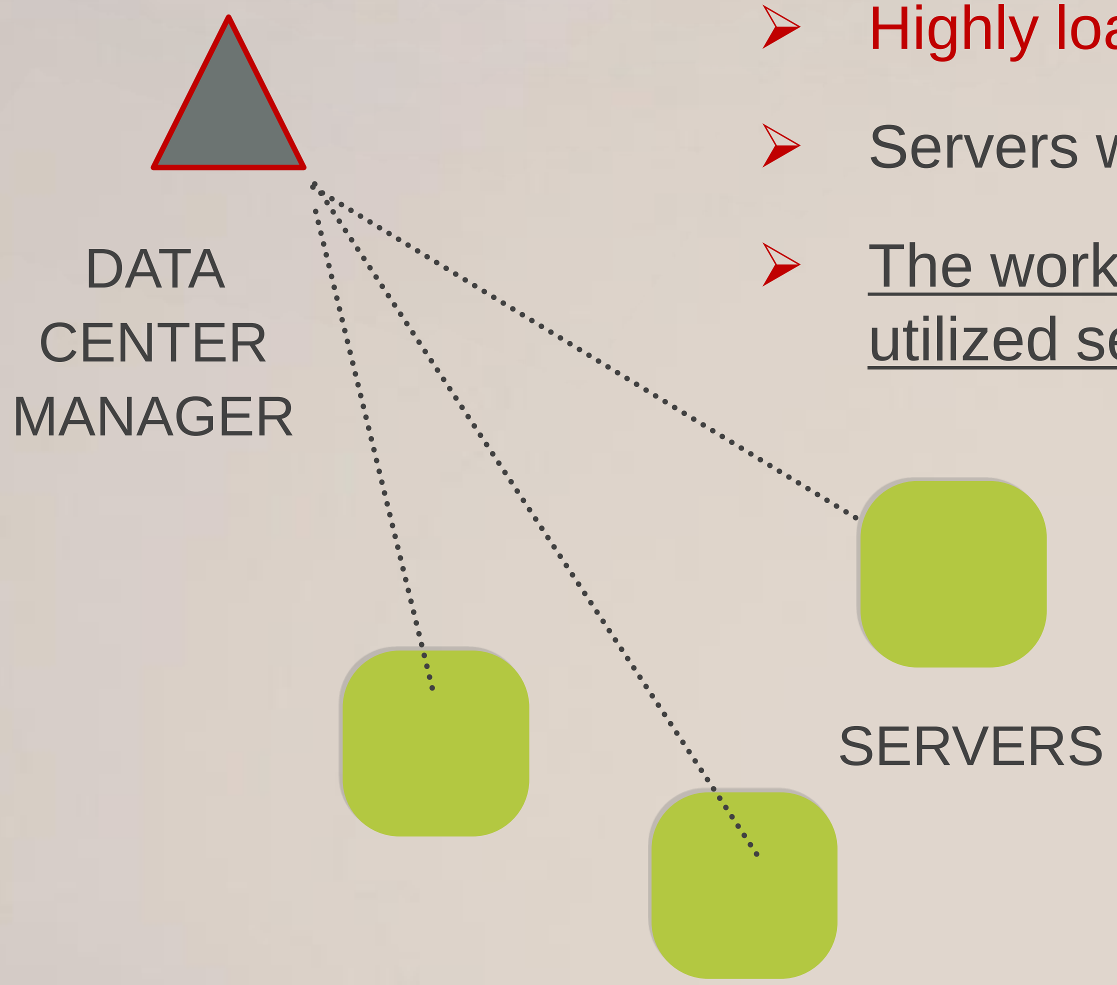
www.eco4cloud.com

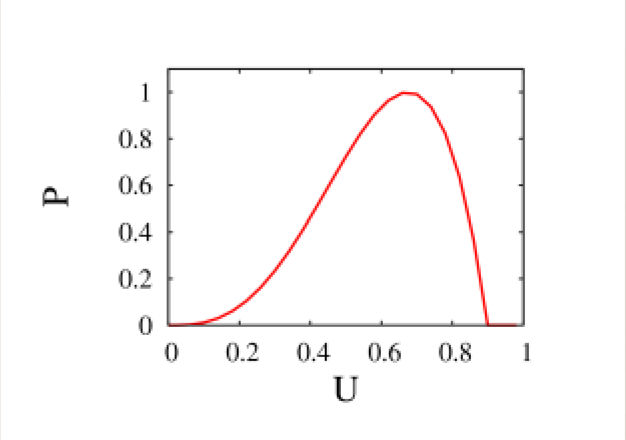
- C. Mastroianni, M. Meo, G. Papuzzo, "Probabilistic Consolidation of Virtual Machines in Self-Organizing Cloud Data Centers". *IEEE Transactions on Cloud Computing*, vol. 1, n. 2, pp. 215-228, 2013.
- PCT Patent "System for Energy Saving in Company Data Centers"
- Licensed to Eco4Cloud, spin-off from National Research Council of Italy

Eco4Cloud in action

The data center manager assigns and migrates VMs to servers based on local probabilistic trials:

- Lightly loaded servers tend to reject VMs
- Highly loaded servers tend to reject VMs
- Servers with intermediate load tend to accept VMs
- The workload is consolidated to a low number of highly utilized servers





Consolidation on distributed data centers

- Many big companies own several data centers
  - Data centers have different and dynamic prices of energy
  - Workload variability both within single sites and across the infrastructure
  - Workload distribution needs to be adapted to reduce costs and improve QoS
- Also important for distributed multi-owner data centers
  - Companies cooperate to gain a bigger portion of the market
  - Users may want to migrate their services among multiple providers

Workload migration among remote DCs

Moving VMs across remote DCs is now possible thanks to:

- Much higher network capacity
- Physical improvements (e.g. wavelength division multiplexing)
- Logical/functional enhancements (e.g., adoption of Software Defined Networks)

Nowadays significant amounts of workload can be moved through dedicated networks or even via regular Internet connections

Workload migration: issues

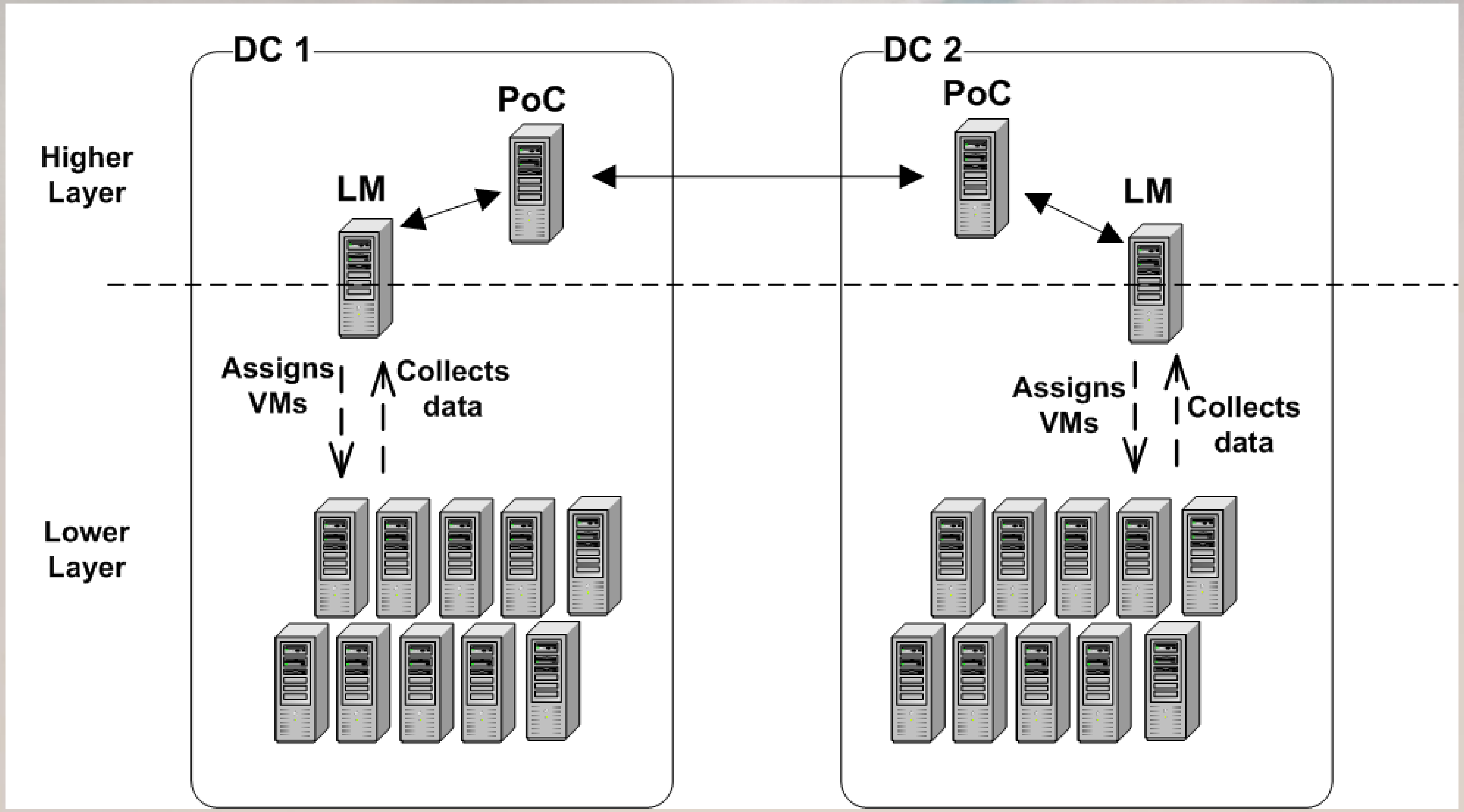
Main questions are:

- *When do the benefits of workload migrations overcome the drawbacks?*
- *From which site to which site to migrate?*
- *Which specific portion of the workload should be migrated?*
- *How to reassign the migrating VMs within the target site?*

Algorithms adopted today try to solve the optimization problem as a whole, originating two main issues:

1) Poor scalability, due to the size of the problem and the large number of parameters and objectives
2) Lack of autonomy: all the data centers must adopt the same strategies and algorithms

Hierarchical architecture for workload distribution



PoC = Point of Contact
LM = Local Manager

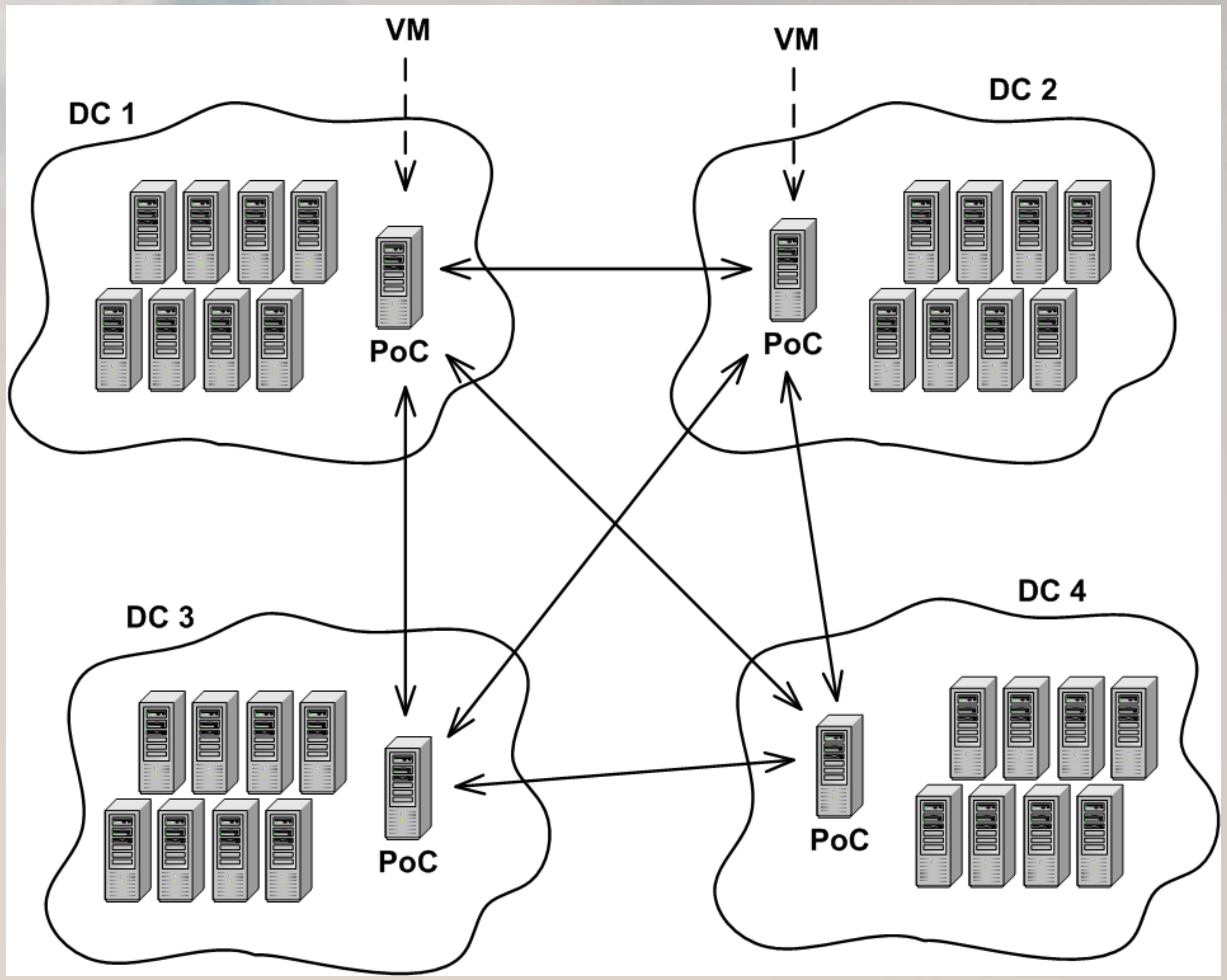
Two layers:

i. the upper layer drives the distribution of VMs among the data centers
ii. the lower layer allocates VMs within single data centers

Scopes/goals of upper and lower layers

The **upper layer**:

i. determines the target data center to which a new VM should be assigned
ii. checks if the workload is well balanced among the different sites
iii. triggers migration of VMs when needed



The **lower layer**:

i. collects information about the state of the local data center, and passes it to the upper layer
ii. assigns VMs internally, with local consolidation algorithms (possibly different from site to site)

Main benefits of the hierarchical architecture

Scalability

→ the size of the problem is reduced by dividing it into two separate problems: inter-DC and intra-DC assignment

Modularity

→ the algorithms of the two layers are independent from each other and can be modified/optimized separately

Autonomy

→ each data center can choose its own local algorithm, depending on technical constraints and management choices

Multi-site assignment algorithm: objectives

When determining to which target DC a VM should be assigned, goals may be as diverse as:

- Reduction of costs

  Costs depend on many factors: power needed for computation, cooling, staff, server maintenance, etc. The cost of energy varies from site to site, and with time

- Reduction of consumed energy
- Reduction of carbon emissions
- Quality of service offered to users
- Load balancing
- Data movement

  Depending on the type of application (data base, Web service) it may be appropriate to assign the VM to the local DC

Goals: carbon emission, load balancing

Representative of two opposite needs: increase efficiency and guarantee fairness among data centers

Each PoC collects two types of information:

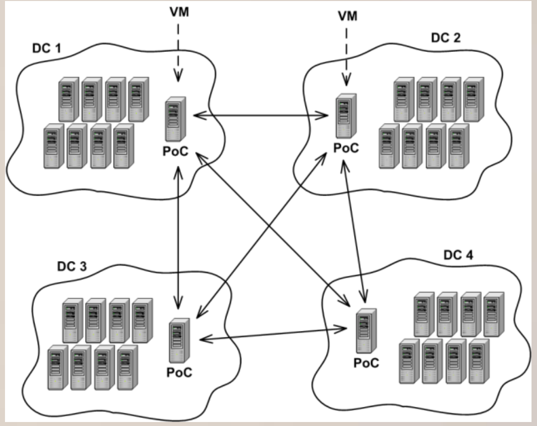
1) **Utilization of resources** in the local data center (CPU, RAM, disk etc.)

2) **Best available carbon footprint rate**

   it is the lowest carbon emission rate of a server available locally.

Each PoC sends/receives this information to/from all the other PoCs

- transmission can be periodic (push) or on request (pull)
- involving a tiny amount of information, just a few bytes…



Selection of the target DC

A new VM is assigned to the data center having the lowest value of the assignment function

$$f^i_{assign} = \beta \cdot \frac{C_i}{C_{max}} + (1-\beta) \cdot \frac{U_i}{U_{max}}$$

For the data center *i*:

- ***Ci*** is the best available carbon footprint rate (multiplied by PUE)
- ***Ui*** is the relative utilization of the bottleneck resource
- Both quantities are normalized with respect to the maximum value

The parameter **β** is used to balance the two goals:

- ***β** = 1: reduce carbon emissions!*
- ***β** = 0: balance the load!*
- *intermediate values of **β**: balance the two goals*

EcoMultiCloud algorithm

Algorithm executed by the upper layer to select the target DC for a new VM

function EcoMultiCloud-AssignmentAlgorithm(β)
    while VM arrives
        for each remote datacenter DC_i
            Request C_i, U_i parameters
        end for
        $C_{max} = Max\{C_i |\ i = 1 \cdots N_{DC}\}$
        $U_{max} = Max\{U_i |\ i = 1 \cdots N_{DC}\}$
        for each DC_i : DC_i is not full, that is, $U_i < U_{T_i}$
            $f^i_{assign} = \beta \cdot \frac{C_i}{C_{max}} + (1-\beta) \cdot \frac{U_i}{U_{max}}$
        end for
        $DC_{target} = DC_j$ such that $f^j_{assign} = min\{f^i_{assign} |\ i = 1 \cdots N_{DC}\}$
        Assign VM to DC_{target}
    end while
end function

Scenario for performance analysis

- 4 data centers, with similar capacities and different values of PUE
- In each data center 112 hosts and 1984 VMs (logs taken from a real case)
- In each data center two *rooms*, with different values of the carbon footprint rate
- The threshold on RAM utilization (bottleneck resource) is 80%

Data center	PUE	Carbon footprint rate (Tons/MWh) Room A	Room B
DC 1	1.56	0.124	0.147
DC 2	1.7	0.350	0.658
DC 3	1.9	0.466	0.782
DC 4	2.1	0.678	0.730

Reference centralized model: ECE

ECE: Energy and Carbon-Efficient VM Placement Algorithm

A. Khosravi, S. Garg, and R. Buyya. Energy and carbon-efficient placement of virtual machines in distributed cloud data centers. Euro-Par 2013

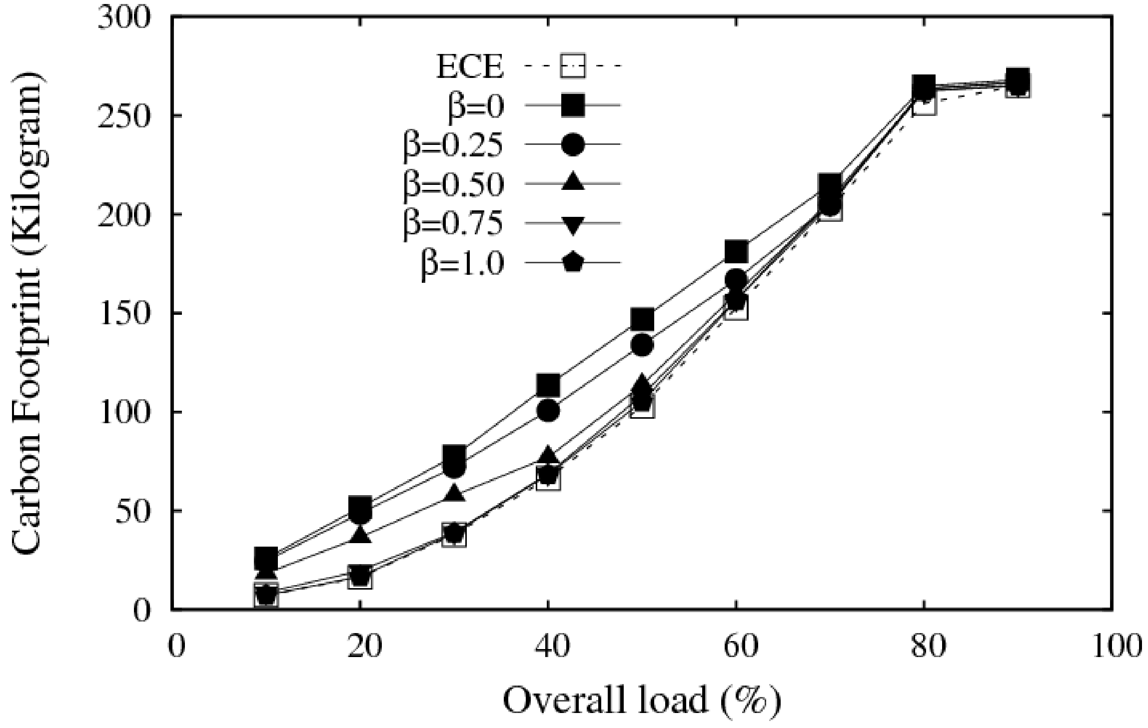
- ECE sorts all the clusters of the multi-DC architecture in ascending order of (PUE x Carbon footprint rate)
- each VM is assigned to the best available cluster and then to the best server in that cluster
- ECE proved to be better than most common heuristics (e.g. First Fit)

Comparison with the hierarchical approach where:

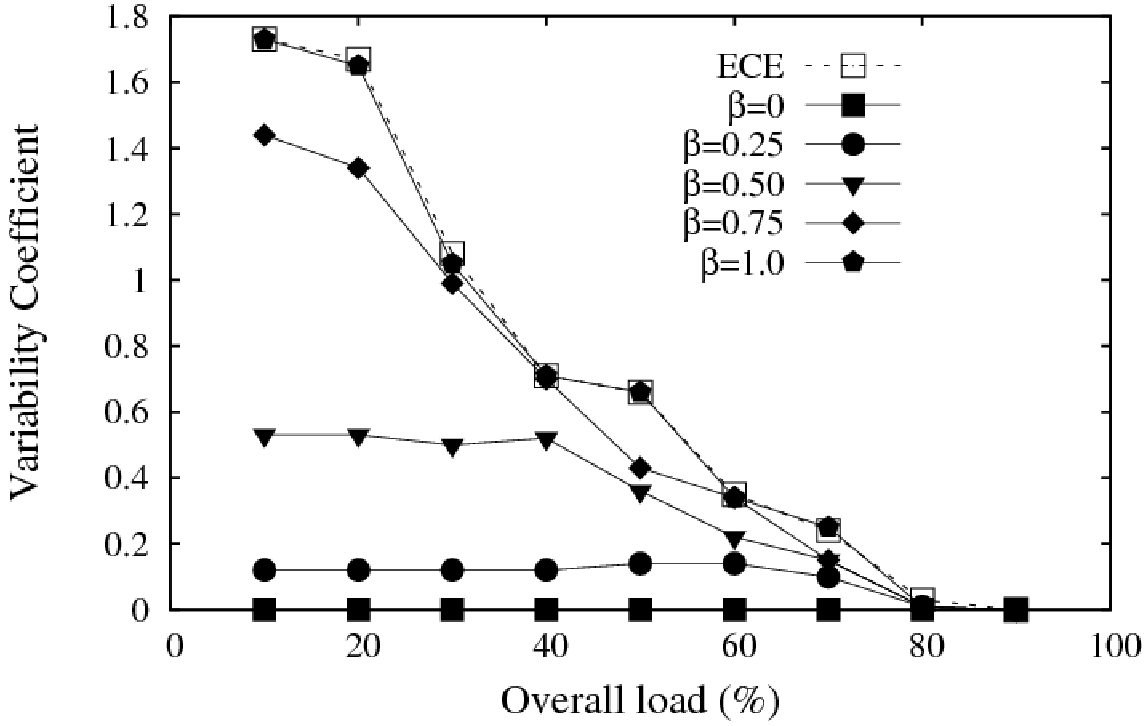
- the **upper layer** uses the **EcoMultiCloud** assignment algorithm
- the **lower layer** uses the ECE algorithm on single DCs (might also use the Eco4Cloud algorithm)

Performances when varying parameter β

Overall carbon footprint (measures efficiency)



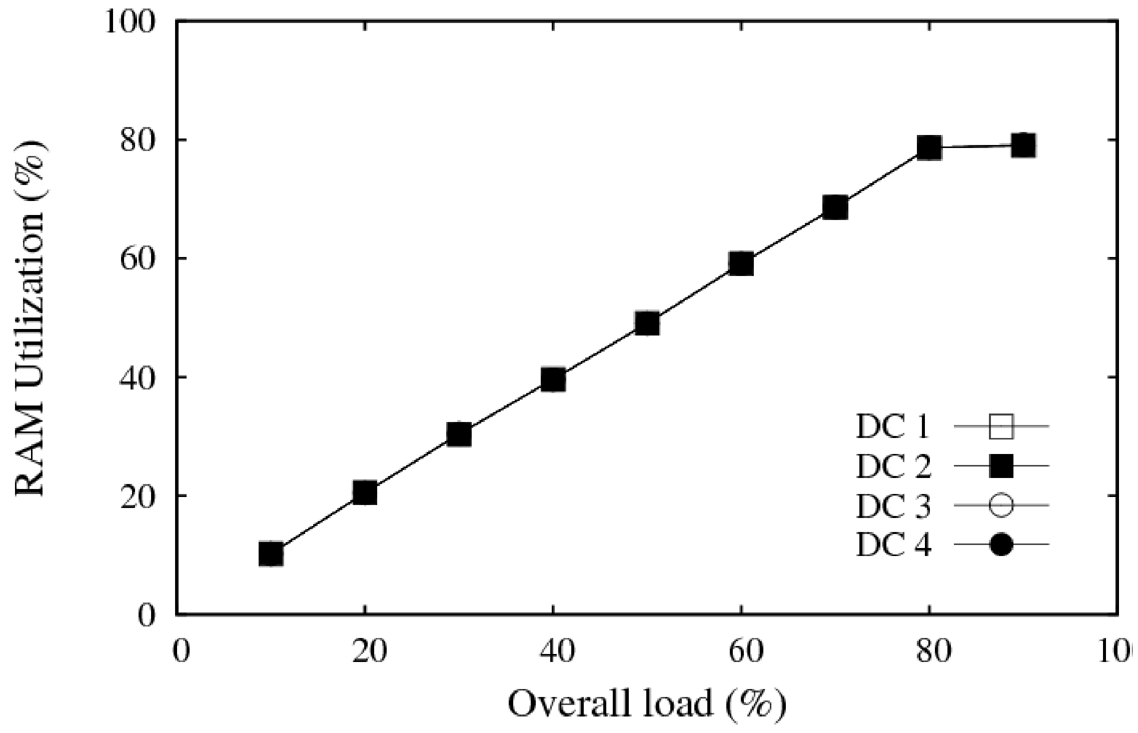
Variability coef. of RAM utilization (measures load balancing)



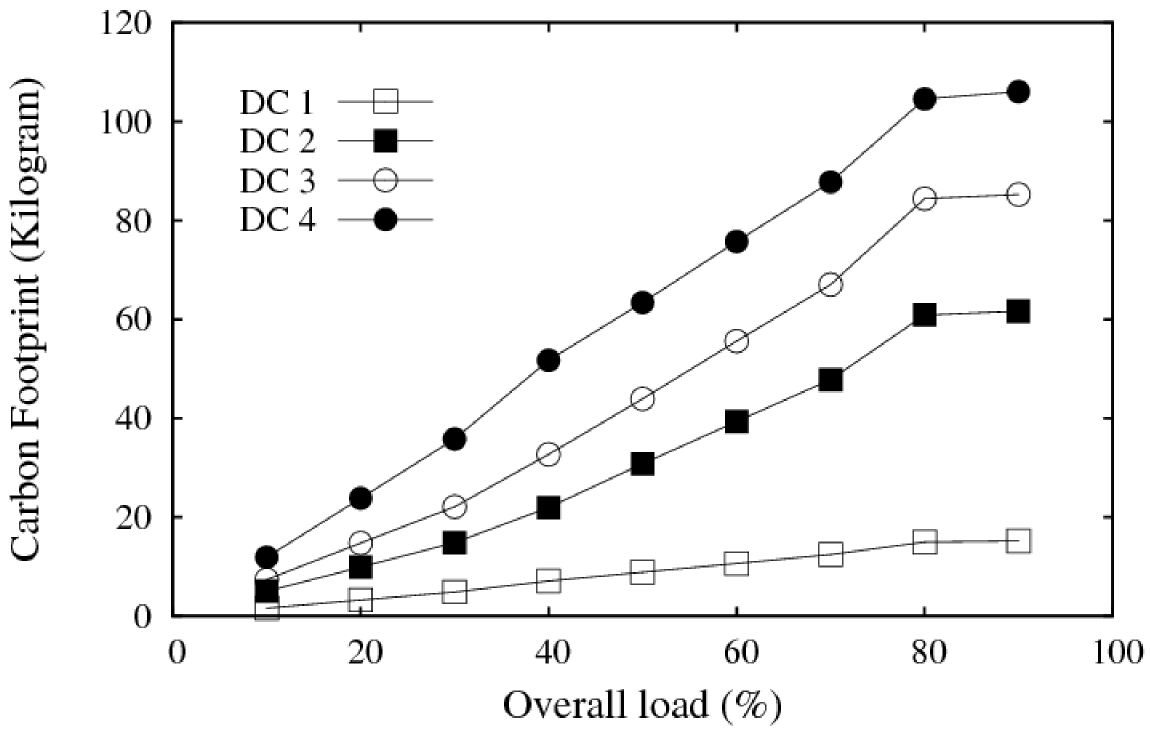
- higher values of **β** allow the carbon footprint to be decreased, at the expense of a greater load imbalance. The proper value should be set in accordance to management requirements
- with **β** = 1, performances are very close to ECE, as load balancing is not considered a goal. This means that the hierarchical approach does not induce any performance degradation

β=0 → maximize load balance

RAM utilization of the 4 data centers



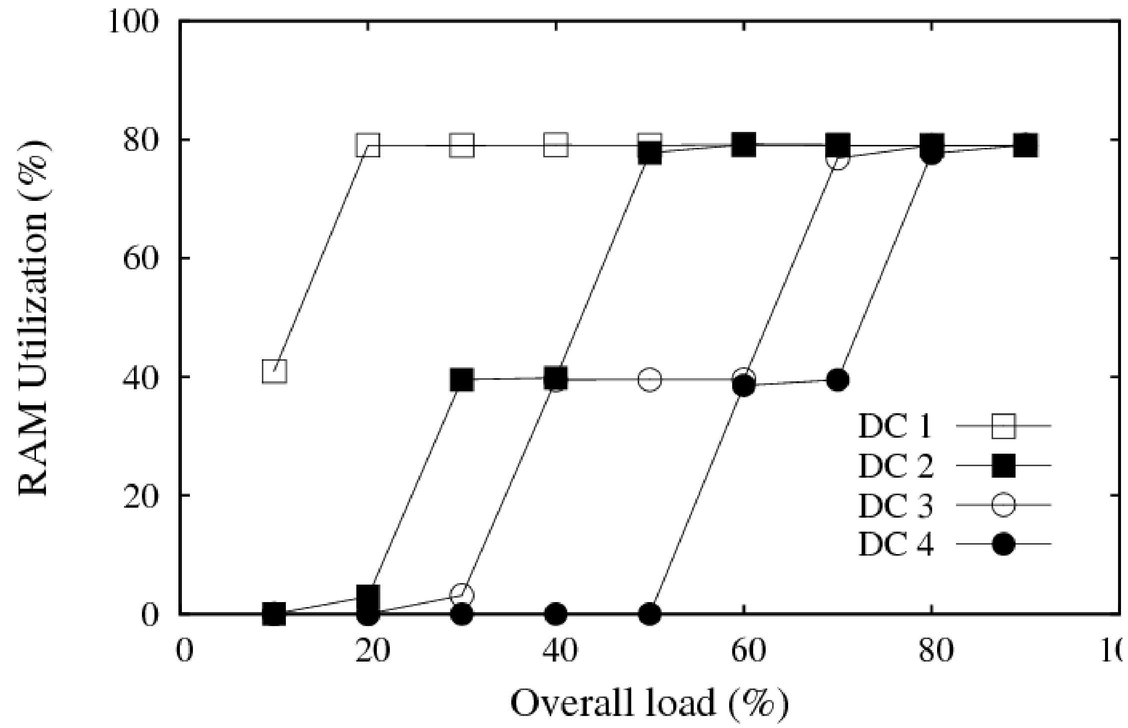
Carbon footprint of the 4 data centers



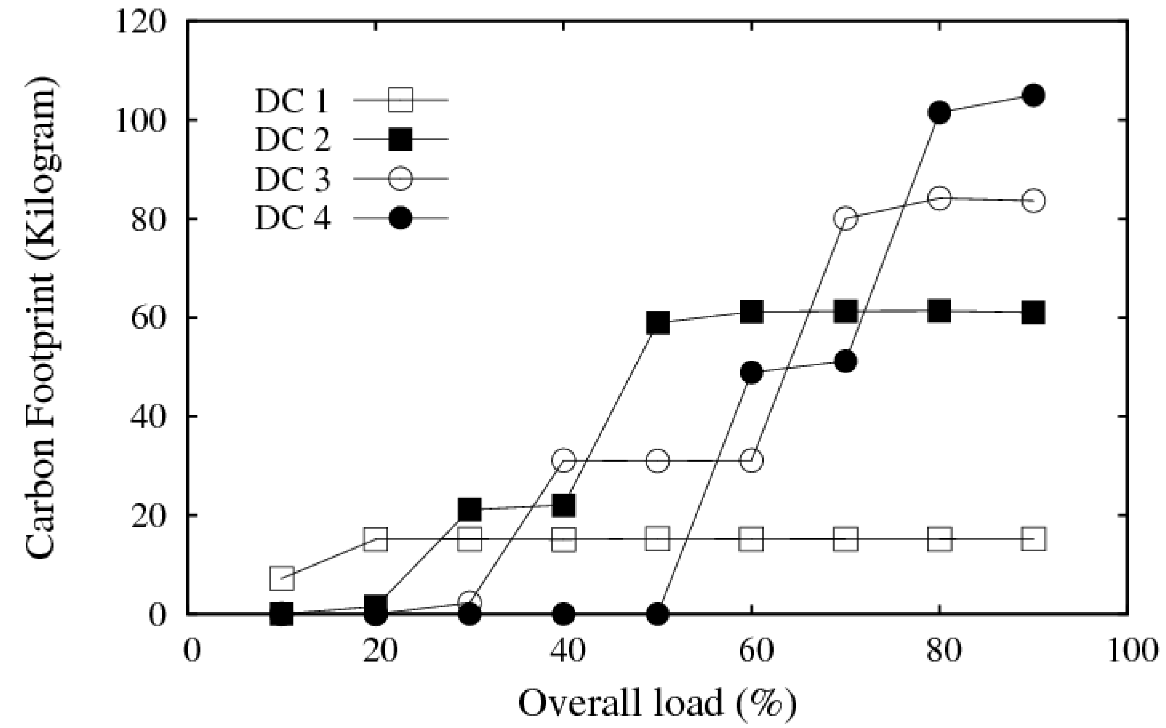
- the data centers have the same RAM utilization: they are loaded at the same rate
- carbon emissions of DCs are proportional to their carbon footprint rates

β=1 → minimize carbon emissions

RAM utilization of the 4 data centers



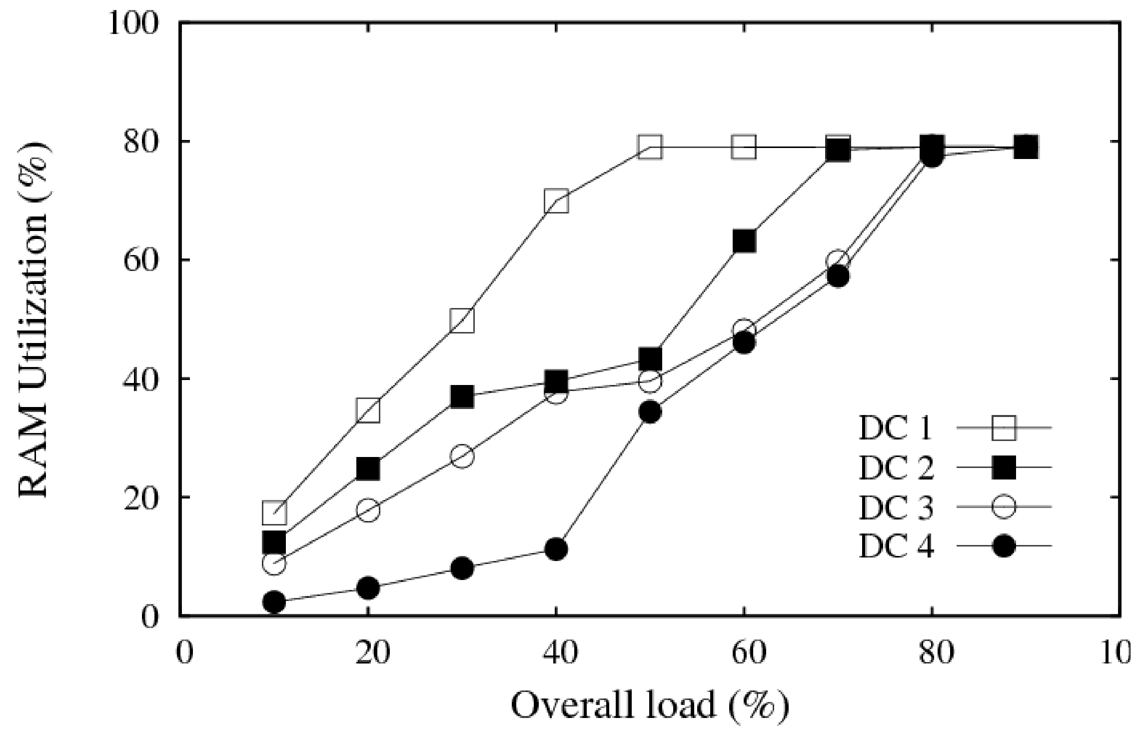
Carbon footprint of the 4 data centers



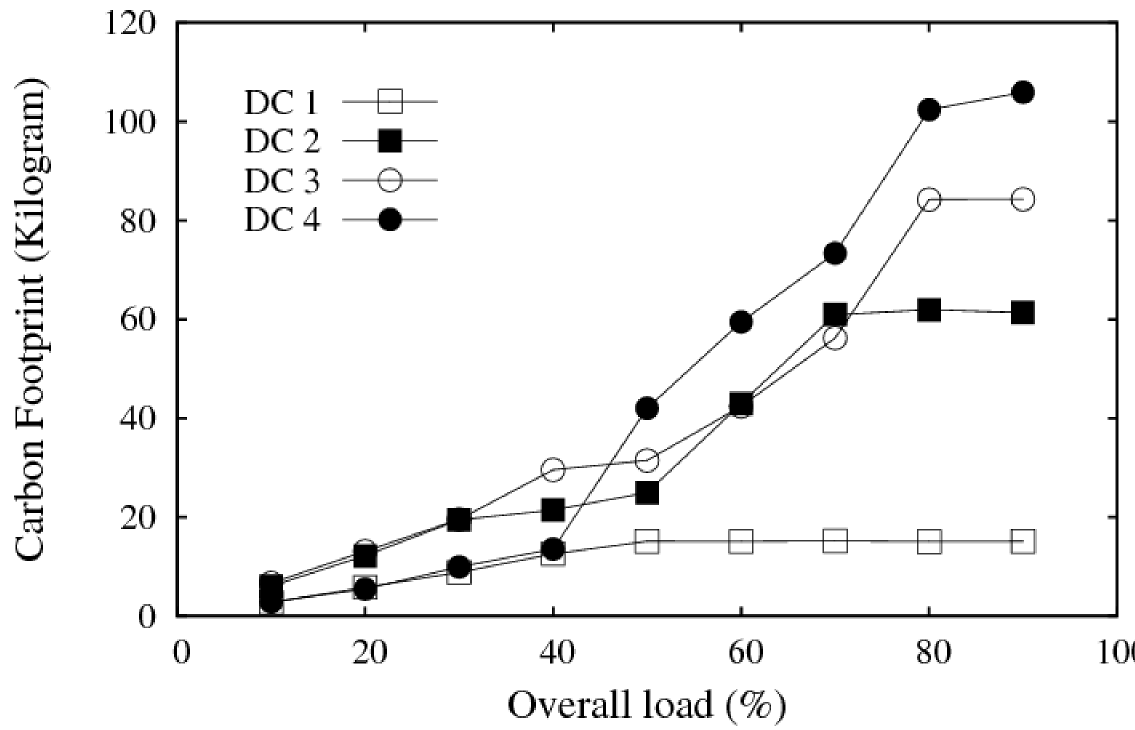
- DC1 is loaded first, the others follow respecting the carbon footprint rates of single rooms
- with low overall load the carbon footprint of DC1 is the highest
- when the load increases, the curves of carbon footprint cross, because less efficient DCs are successively loaded

β=0.5 → balance the two goals

RAM utilization of the 4 data centers



Carbon footprint of the 4 data centers



- the load is assigned to the DCs with different rates
- values of carbon emissions depend both on the efficiency of DCs and on their utilization

Mathematical Analysis

At the steady-state, all DCs has the same value of the assignment function

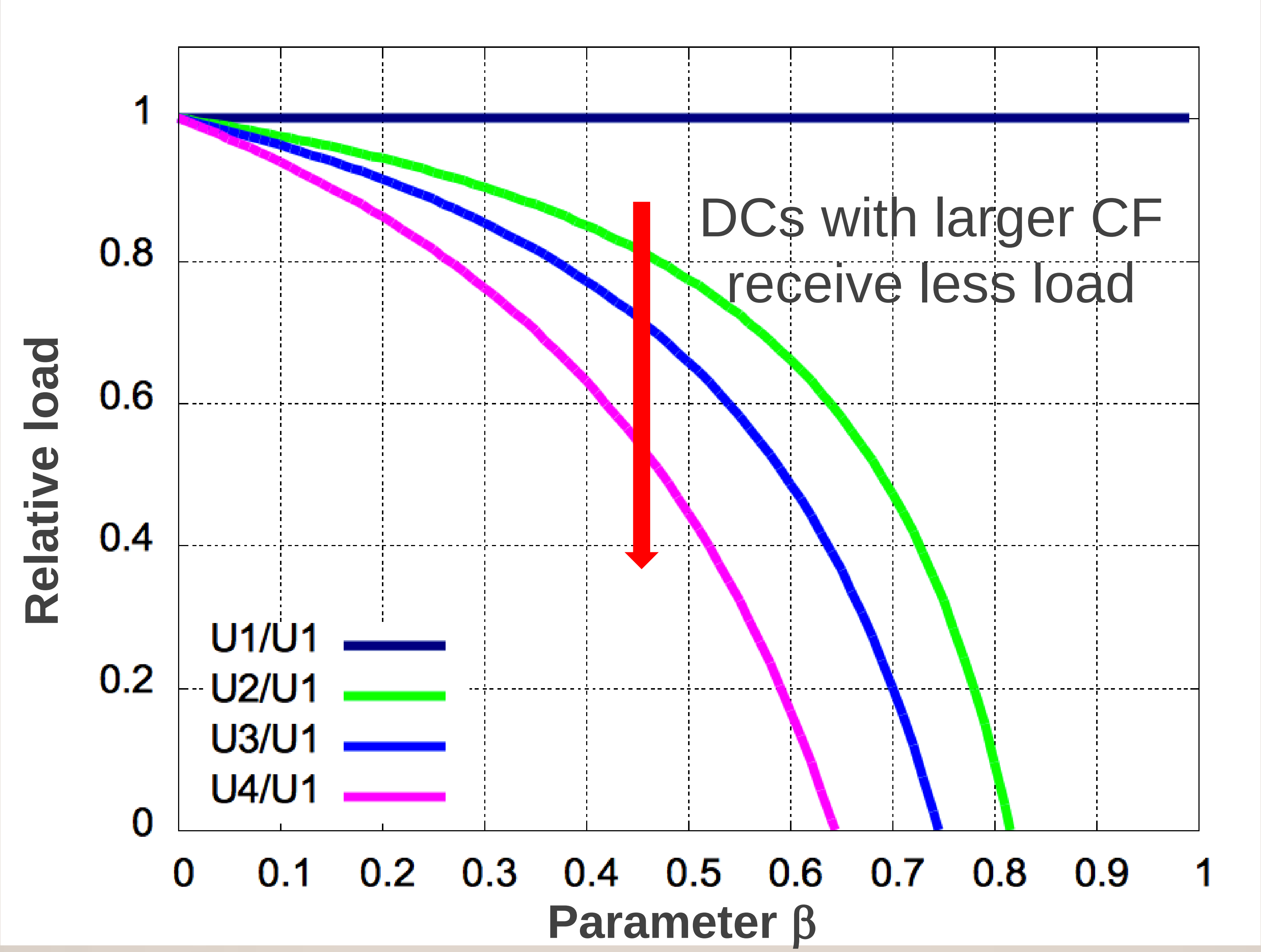
The steady state distribution of the total load **Λ** over the DCs is then given by the set of equations

$$\begin{cases} \beta \dfrac{C_i}{C_{\max}} + (1-\beta)\dfrac{U_i}{U_{\max}} = \beta \dfrac{C_j}{C_{\max}} + (1-\beta)\dfrac{U_j}{U_{\max}} \\ \sum_{i=1}^{D} U_i = \Lambda \end{cases}$$

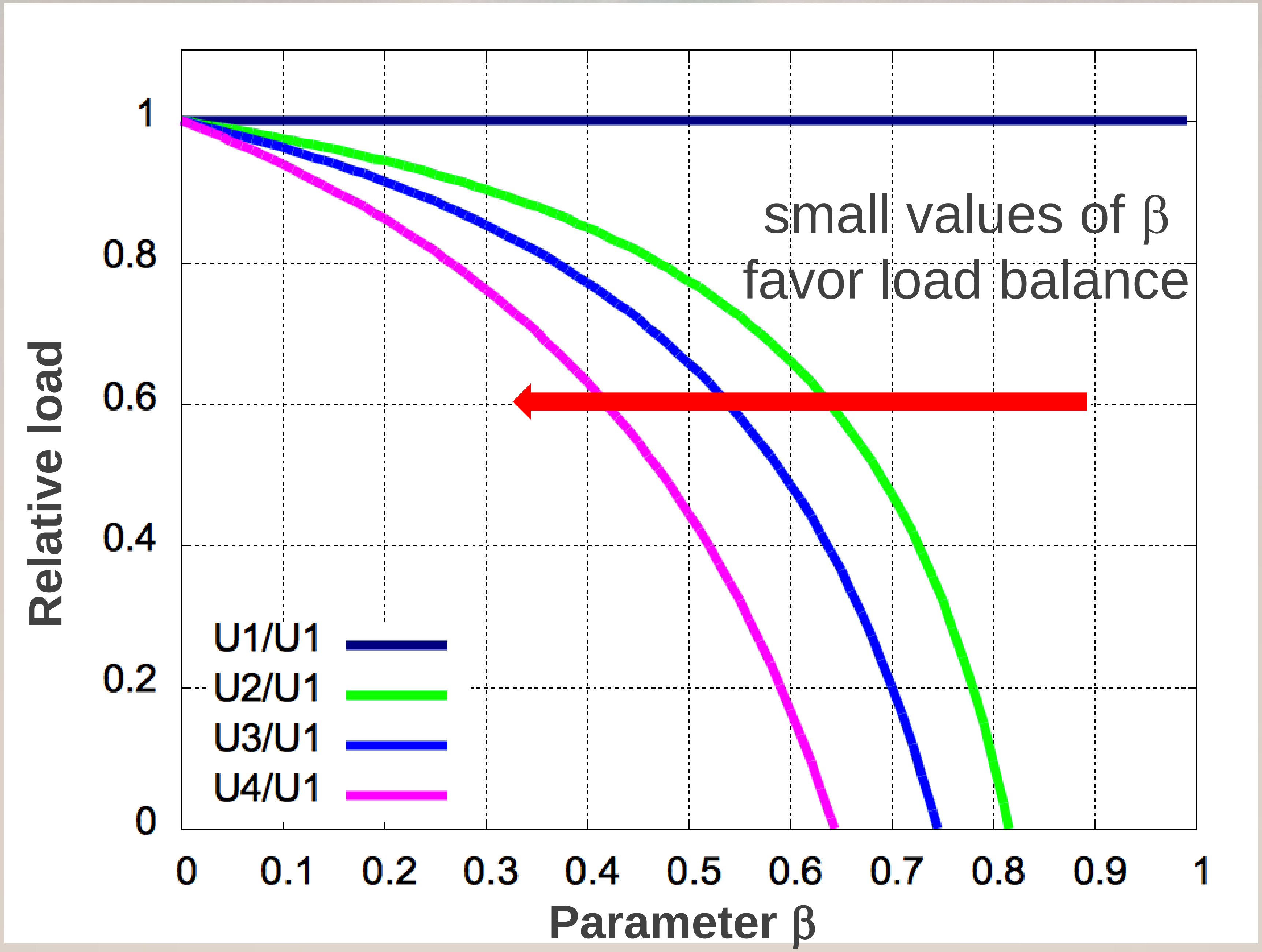
The solution of this set of equations can be used to optimize the value of **β**

Distribution of load vs. the value of β

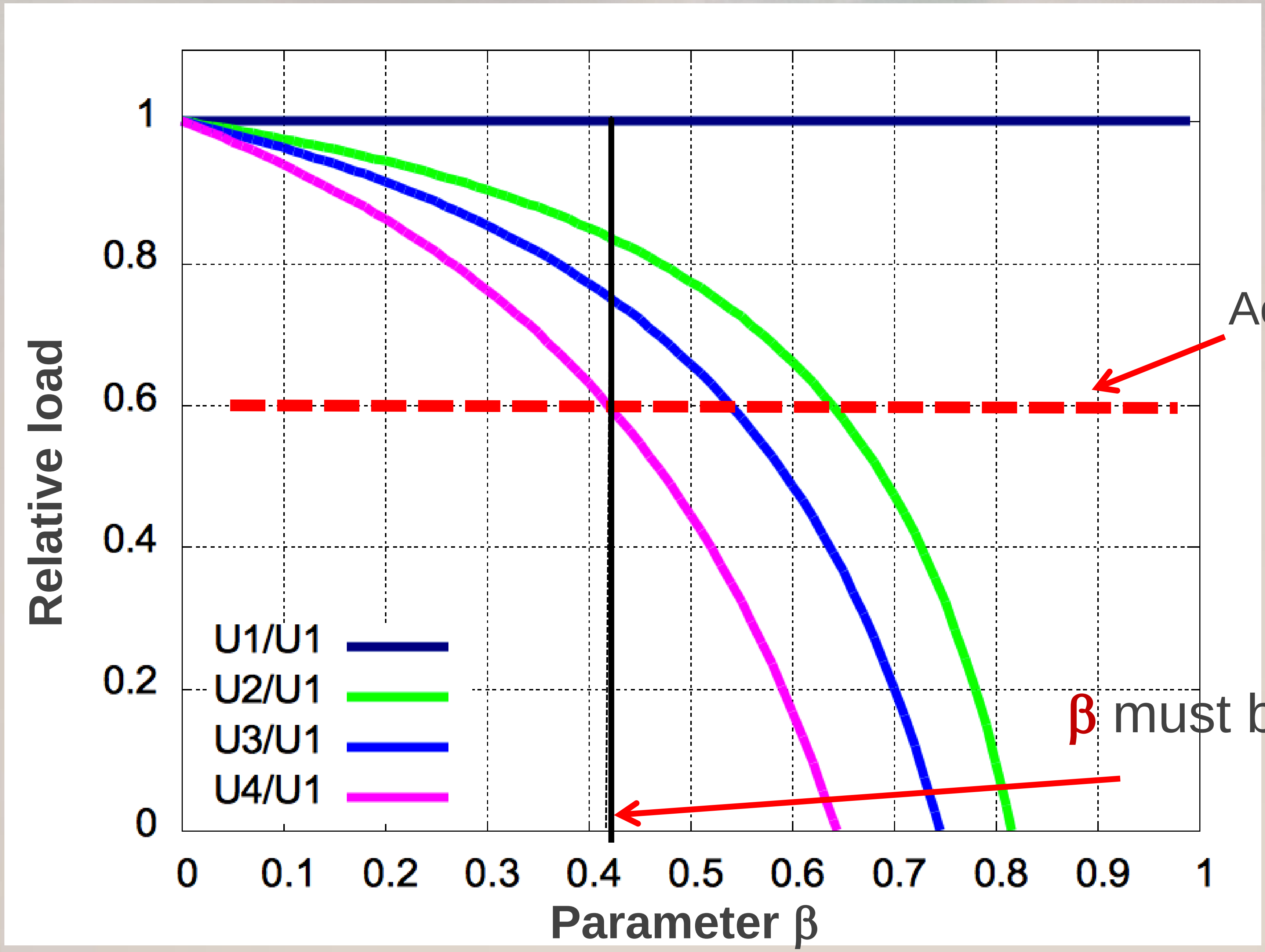
Relative load of DCs w.r.t. the load of DC1, which is always the most loaded



Load balance vs. the value of β



Optimization with constraint on load balance



Acceptable load imbalance:
relative load above 0.6

β must be below this value

Conclusions

- New hierarchical approach for green workload distribution in a multi-DC scenario
- Benefits: scalability, modularity, autonomy of single DCs (and no performance degradation w.r.t. centralized algorithms)
- Tunable to balance different objectives (costs, carbon emissions, load balance, etc.)
- Easy to analyze mathematically, which helps to optimize performances while matching given constraints

Next steps

- Algorithms for dynamic workload migration, for example in presence of time-dependent energy prices

THANK YOU!

Carlo Mastroianni

ICAR-CNR & eco4cloud srl

Rende (CS) Italy

--

www.eco4cloud.com

mastroianni@eco4cloud.com

fb: www.facebook.com/eco4cloud