Scheduling on heterogenous architectures to optimize HPC Data Center energy consumption

Power proportionnality

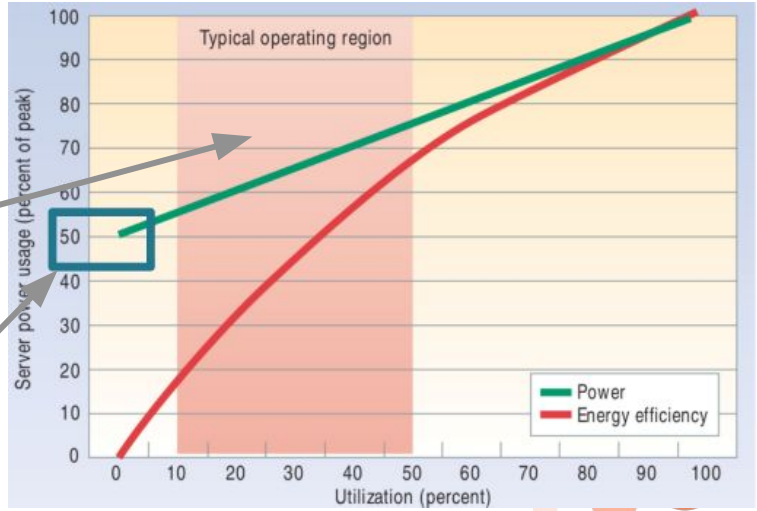
Some facts

Data-centers are over-dimensioned

- Servers are rarelly fully loaded (average load between 10 and 50%)
- Some servers are sometimes never utilized

[The case for Energy-Proportional Computing, L.A. Barroso and U. Hölzle, IEEE Computer, 2007]

- Servers' typical operating region: between 10 and 50 % utilization
- High static power consumption: poor energy efficiency at low utilization



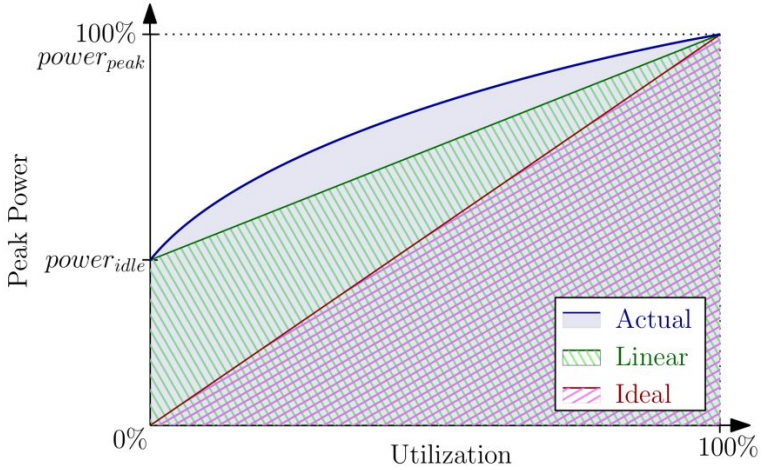
Build an Energy Proportional infrastructure with Non Energy Proportional servers

Violaine Villebonnet PhD Thesis, 2016

The goals:

- no static, only dynamic consumption
- consumption proportional to utilization
- constant energy efficiency over utilization

=> at cluster level: adapt the composition of infrastructure with dynamic provisioning



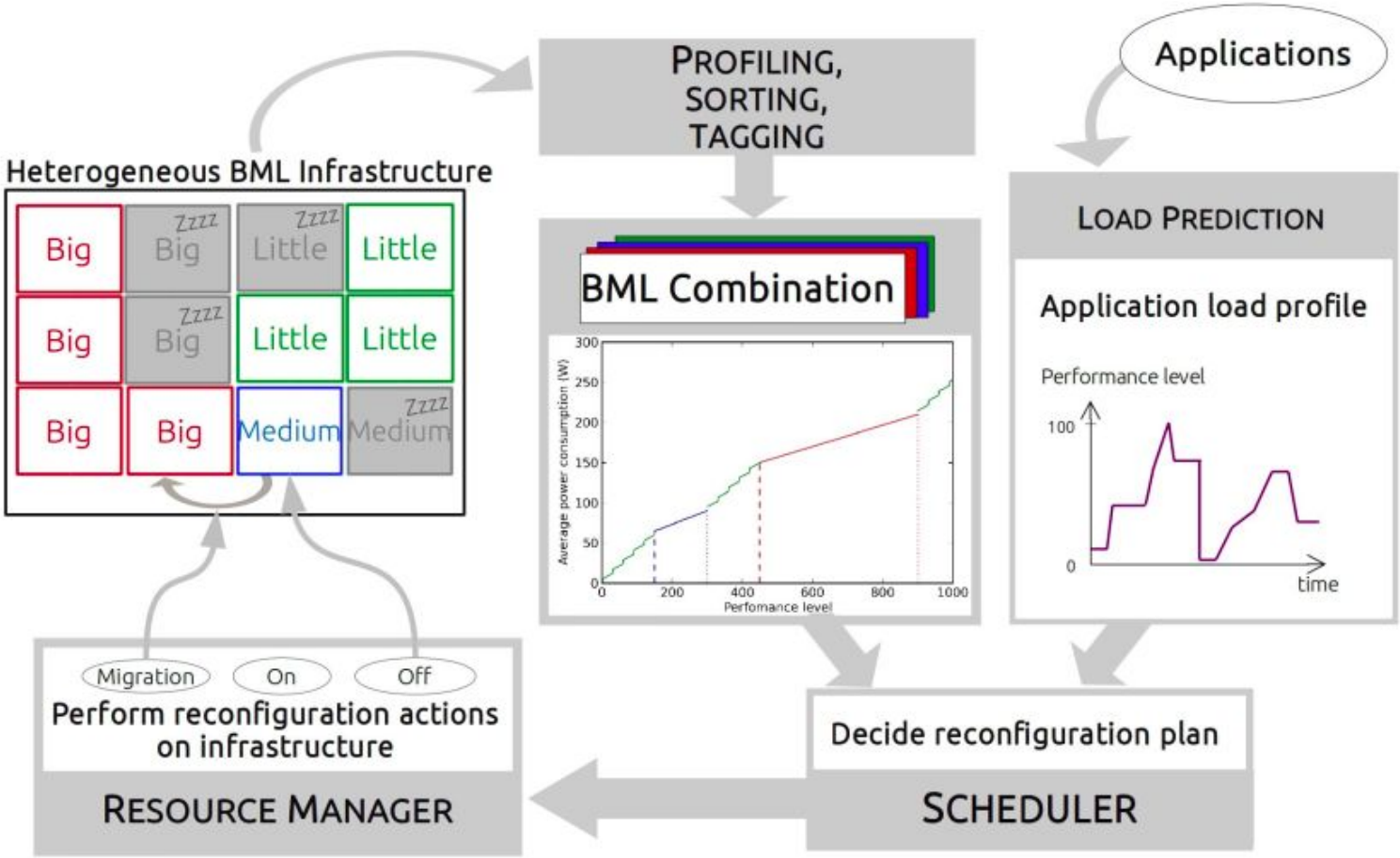
“Big Medium Little” Data center infrastructure

- Heterogeneous Infrastructure composed of different types of computing resources chosen for their performance and energy consumption characteristics

Samsung Chromebook
ARM Cortex-A15 (2 coeurs)

Raspberry PI 2
ARM Cortex-A7 (4 coeurs)

- Reach Energy Proportionality by always using only the most energy efficient (combinations) of machine(s) for the current performance needs of the application


Heterogeneous BML Infrastructure
Big
Big
Zzzz
Little
Little
Big
Big
Zzzz
Little
Little
Big
Big
Medium
Medium
Zzzz
PROFILING, SORTING, TAGGING
Applications
BML Combination
Average power consumption (W)
Perfomance level
LOAD PREDICTION
Application load profile
Performance level
100
0
time
Migration
On
Off
Perform reconfiguration actions on infrastructure
RESOURCE MANAGER
Decide reconfiguration plan
SCHEDULER

Ideal BML scheduler

- Predict future load level (Perfect knowledge over a sliding look-ahead window)
- Compute Ideal BML combination for this load level
- If this ideal combination is different from current configuration :

  Perform reconfiguration towards the new combination

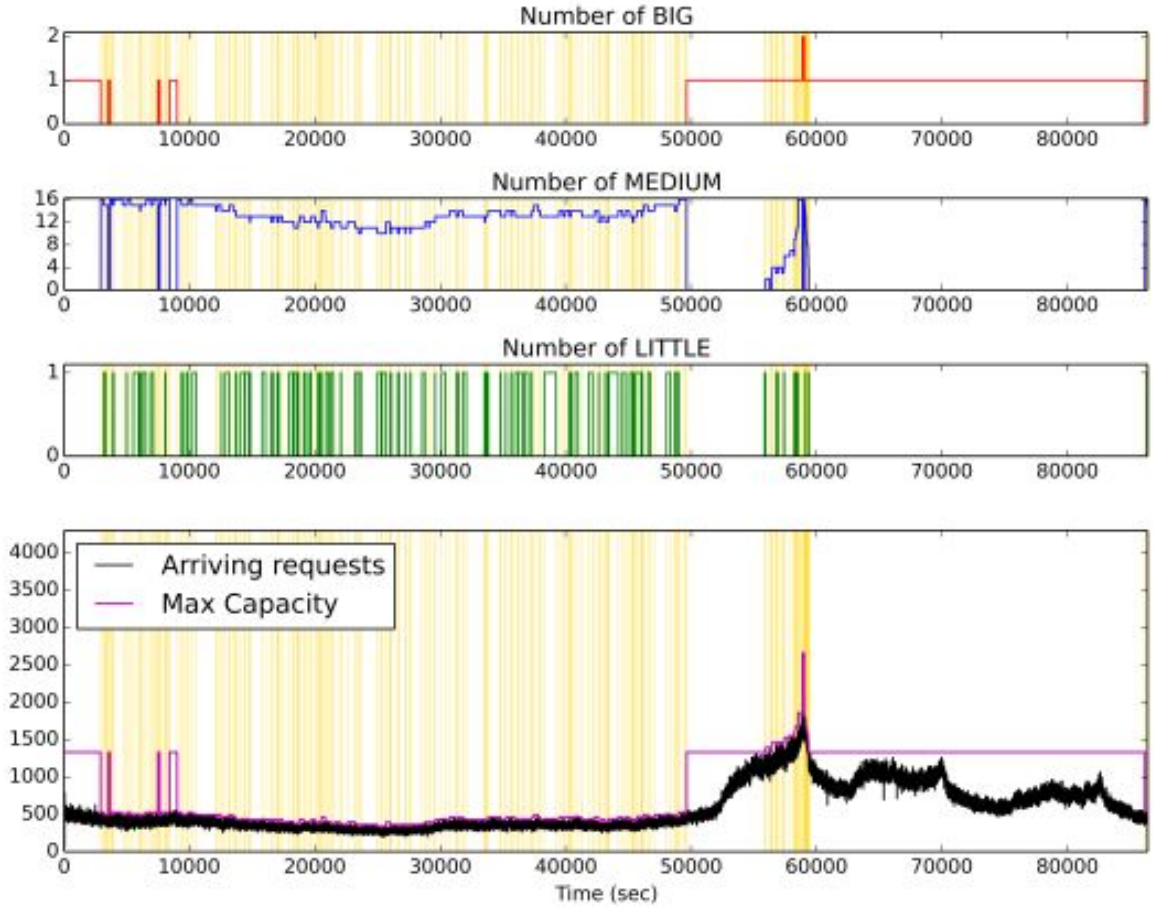
  → (First Switch On, then Switch Off actions)

  Once reconfiguration completed (All On and Off actions ended), New look-ahead window starts
- Else new look-ahead window starts at next time step (1second)

V. Villebonnet, G. Da Costa, L. Lefèvre, JM. Pierson, P. Stolf
Energy Proportionality in Heterogeneous Data Center Supporting Applications with Variable Load.
IEEE International Conference on Parallel and Distributed Systems, ICPADS, 2016.

Traces: 1998 WorldCup web site access log

Day 48 of WorldCup'98 Traces



Perf BML :

- Joules per request: 0,2152
- Infrastructure utilization: 70,1%
- Number of reconfigurations: 221

Multi-Terms Algorithm

At each time step, the algorithm proposes first:

- Load-Reactive Actions:

- Switch-On if future load increases

- Switch-Off if future load decreases

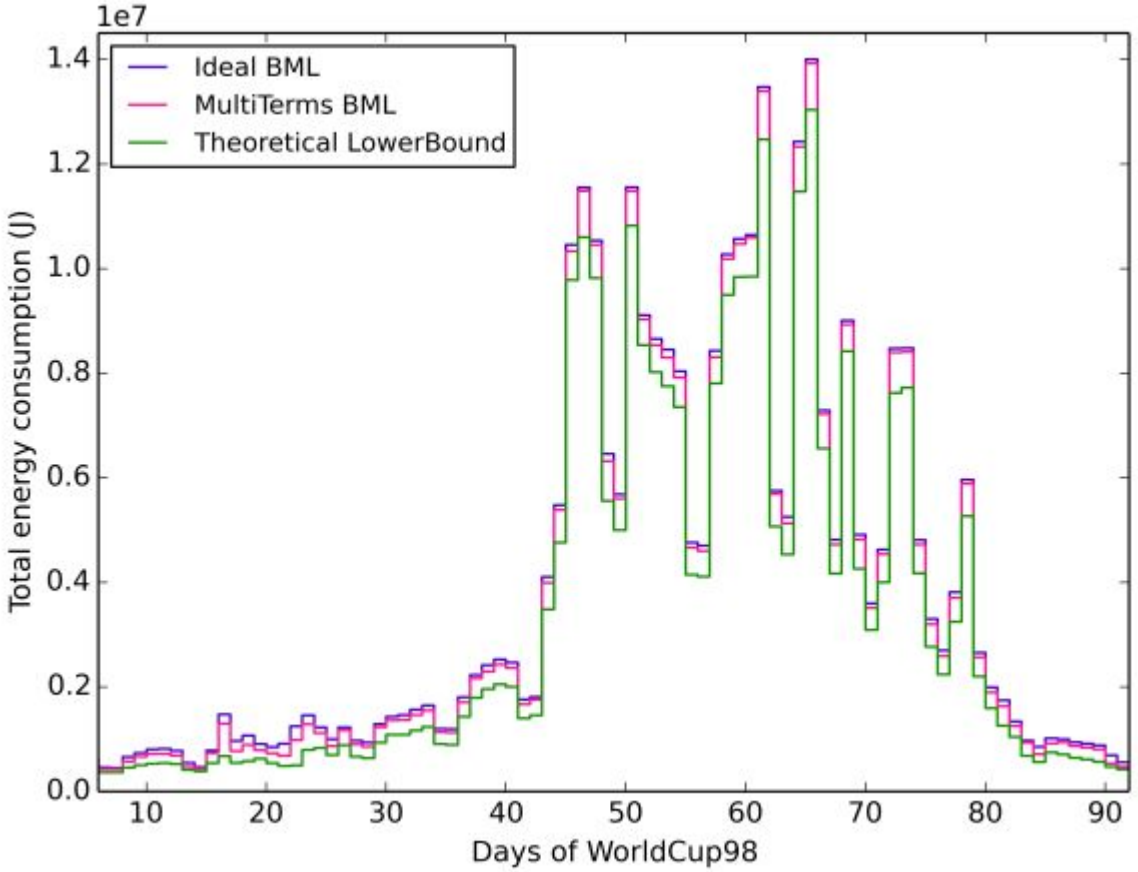
If no load-reactive actions are needed, the algorithm can perform:

- Energy-Saving Actions: Back-to-Ideal

⇒ It's necessary to observe different moments of the future to take reconfigurations decisions → different look-ahead windows

Multi-Terms provisioning algorithm always consumes less

than Ideal BML



REF SBAC PAD V. Villebonnet, G. Da Costa, L. Lefèvre, J-M. Pierson, P. Stolf, SBAC-PAD 2016

Other algorithms

- V. Villebonnet, G. Da Costa, L. Lefèvre, JM. Pierson, P. Stolf

Energy Aware Dynamic Provisioning for Heterogeneous Data Centers. International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD 2016)

- V. Villebonnet, G. Da Costa, L. Lefèvre, JM. Pierson, P. Stolf

Dynamically Building Energy Proportional Data Centers with Heterogeneous Computing Resources. IEEE International Conference On Cluster Computing (CLUSTER 2016), p. 217-220.

Thermal-aware scheduling

Cooling

Some facts

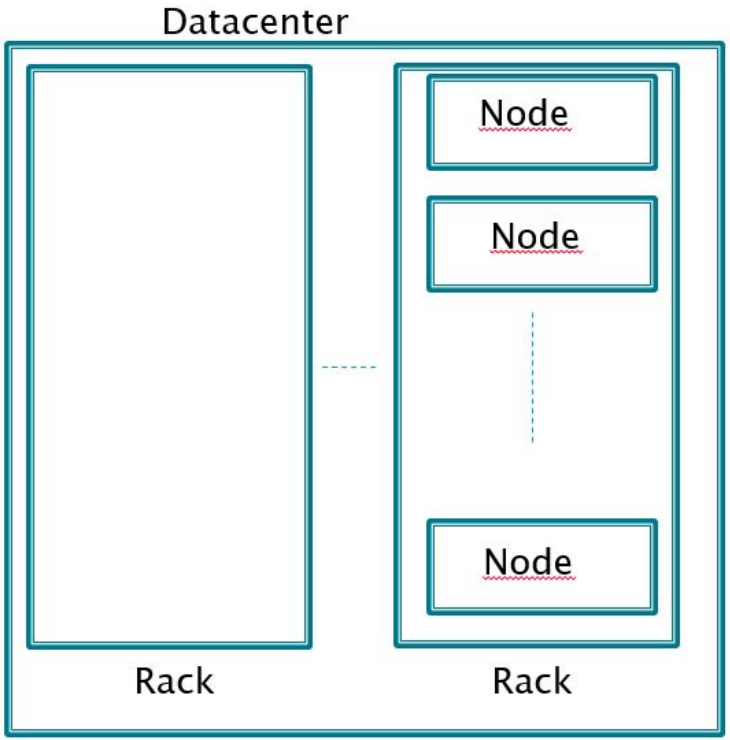
- Processor load impacts energy consumed -> generates heat + CRAC consumption
- Cooling represents around 40% of DC power consumption (as IT equipment)
- High temperature is undesirable in the operation of a datacenter because:
    - it reduces the reliability of the servers
    - it induces a larger cooling cost
    - high temperature increases static power consumption
- Project CoolEmAll, post doc Hongyang Sun.

Three thermal-aware scheduling possibilities

- At Datacenter level: considering global energy cost (cooling + compute)

$$P^{cool}(t) = \frac{\sum_{j=1}^{m} P_j^{comp}(t)}{\mathrm{CoP}(T^{sup}(t))}$$

$$\mathrm{CoP}(T) = 0.0068T^2 + 0.0008T + 0.458$$



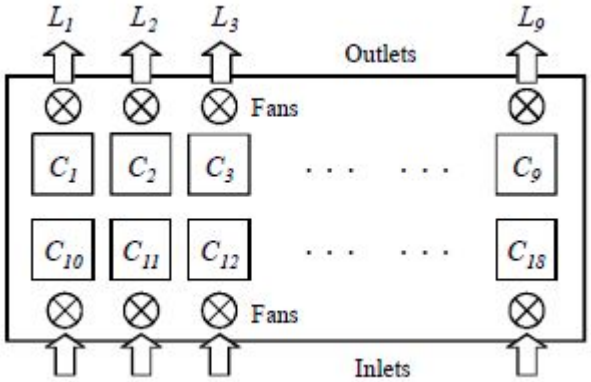
H. Sun, P. Stolf, JM. Pierson, G. Da Costa
Energy-efficient and thermal-aware resource management for heterogeneous datacenters. Sustainable Computing: Informatics and Systems, Elsevier, Vol. 10.1016/j.suscom.2014.08.005, p. 1-15, août 2014.

- At rack level:

Different heuristics to handle t° between sets of nodes: multi-objectives, Coolest

Considering heat distribution

Spatial temperature model



$$T_i^{in}(t) = T_{sup} + \sum_{k=1}^{m} d_{i,k} \cdot P_k(t)$$

$$d_{x,k} = \begin{cases} 1, & \text{if } x = k \\ 0.84, & \text{if } x = k+9 \\ 0, & \text{otherwise} \end{cases}$$

Tang, 2008

Leandro Fontoura and al.,Cooling energy cost Energy-Efficient, Thermal-Aware Modeling and Simulation of Datacenters: The CoolEmAll Approach and Evaluation Results, Ad Hoc Networks Journal, 2015

At node level

Temporal and spatial model

$$T_i(t) = (1-f)\left(P_i(t)R + T_{sup} + \sum_{k=1}^{m} d_{i,k}P_k(t)\right) + fT_i(t-1) \qquad f = e^{-\frac{1}{RC}}$$

Optimization of performance (makespan) and temperature. Heuristics with two steps:

1. Job assignement with load balancing policy
2. Thermal management: speed of each server (reduce the speed, idle but no interleaving) to respect T thresh

Linear programming resolution, comparison with heuristic

JM. Pierson, P. Stolf, H. Sun, H. Casanova. MILP formulations for spatio-temporal thermal-aware scheduling in Cloud and HPC datacenters. Cluster Computing 2019.

H. Sun, P. Stolf, JM. Pierson. Spatio-temporal thermal-aware scheduling for homogeneous high-performance computing datacenters. FGCS, 2017.

Malleable tasks

http://www.irit.fr/energumen

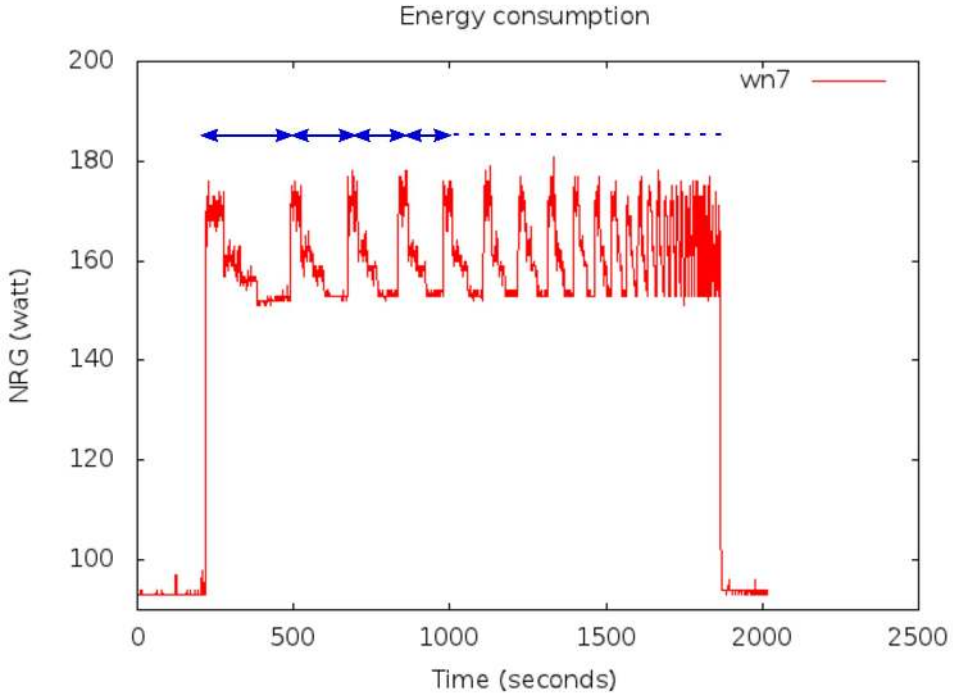
ANR-Energumen project

Dupont, Briag, Nesryne Mejri, and Georges Da Costa. "Energy-aware scheduling of malleable HPC applications using a Particle Swarm optimised greedy algorithm." IGSC 2020.

Classical iterative application

Iterative Schwarz algorithm

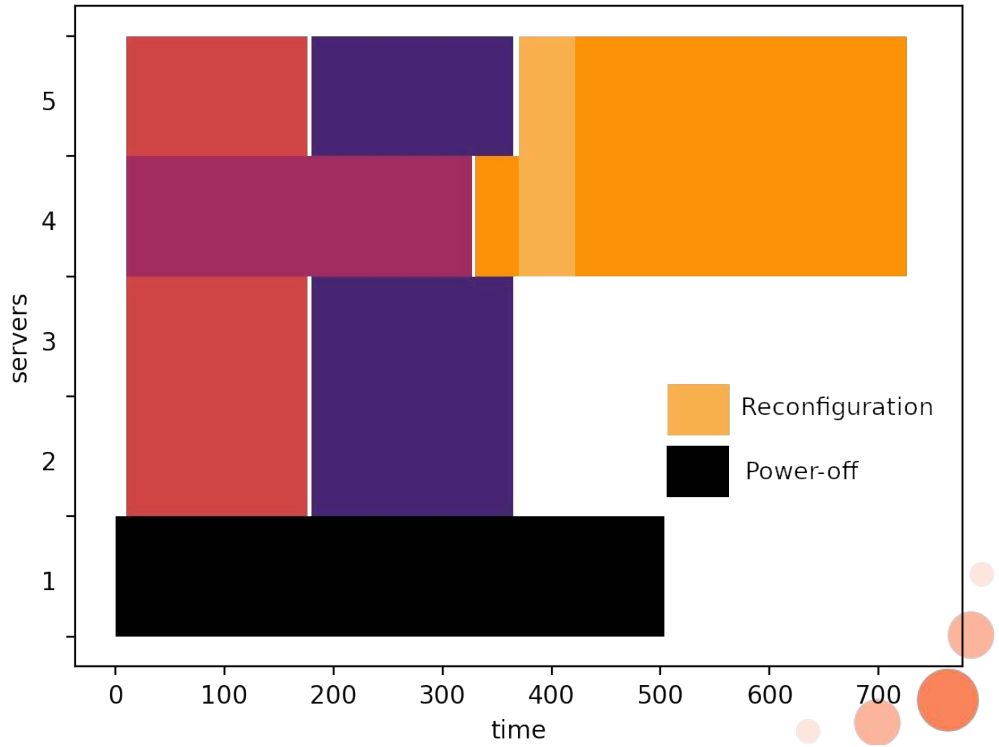
- Start with computing
- Finishes with mostly communication
- Number of servers
  - Adapted for first phases
  - Slow-down the final phases



Malleable tasks

New capability:

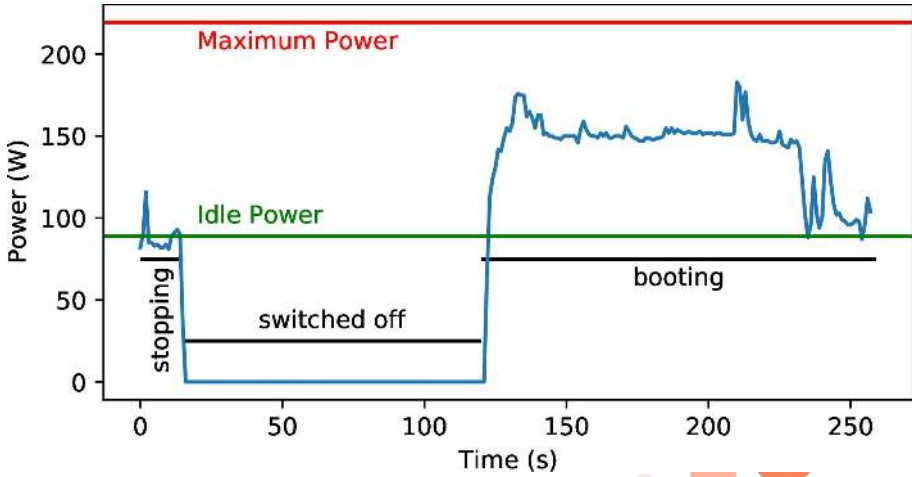
- Change resource allocation
  - At starting time
  - During execution
- Need code source modification



Decisions are based on predictions

Similar concept as ON/OFF

- To compensate for stopping and booting time:
  - Prediction of idle time
- To compensate for reconfiguration time:
  - Prediction of remaining time
  - Prediction of speed-up



Objective

Focus on Scheduling

1. Scheduling proposals
   a. Existing theoretical proof of optimality
   b. Complexity evaluation of greedy algorithms
   c. Objectives: Performance and Energy

2. Use of BatSim (based on SimGrid)
   a. Scheduler
   b. Results can be directly ported in SLURM/OAR

```
int GetCount(int n, int v[])
{
    int total_count = 0;
    int count = 0;
    int id;         // process id, i.e. "r
    int p;          // number of concurren
    int low, high;  // this rank's partiti

    MPI_Comm_rank(MPI_COMM_WORLD, &id);
    MPI_Comm_size(MPI_COMM_WORLD, &p);

    low = id*(n-1)/p;
    high = (id+1)*(n-1)/p;

    for (int i = low; i < high; i++)
    {
        count += v[i];
    }

    if (p > 1)
        MPI_Reduce(&count, &total_count, 1
    else
        total_count = count;

    return total_count;
}
```