Modelling the EGEE latency to optimize job performances

Diane Lingrand, Tristan Glatard, Johan Montagnat
CNRS, I3S Laboratory

Motivations

- **Jobs submitted to a grid are impacted by a latency coming from:**
  - Submission time
  - Scheduling time
  - Queuing time
  - Unrecovered faults (hardware / software)
- **This result in a significant overhead**
  - Single penalty per job
  - The shorter the job, the higher the relative overhead
- **Many grid application environments today use pilot jobs**
  - Reduces the number of *submitted* jobs and therefore the overhead
- **An alternative solution is to find the optimal submission parameters given the infrastructure load**

The overhead depends on the system scale and load

- **Grid'5000 clusters (at the time of the experiments)**
  - Multi-cluster computing resources
    - Sophia cluster: 105 nodes
    - Grenoble cluster (IDPOT): 21 nodes
  - Storage resources: NFS mounted home directories
  - Connectivity: Local Area Network

- **EGEE production grid (at the time of the experiment)**
  - Computing resources: ~20 000 nodes
  - Storage resources: network Storage Elements
  - Connectivity: Wide Area Network

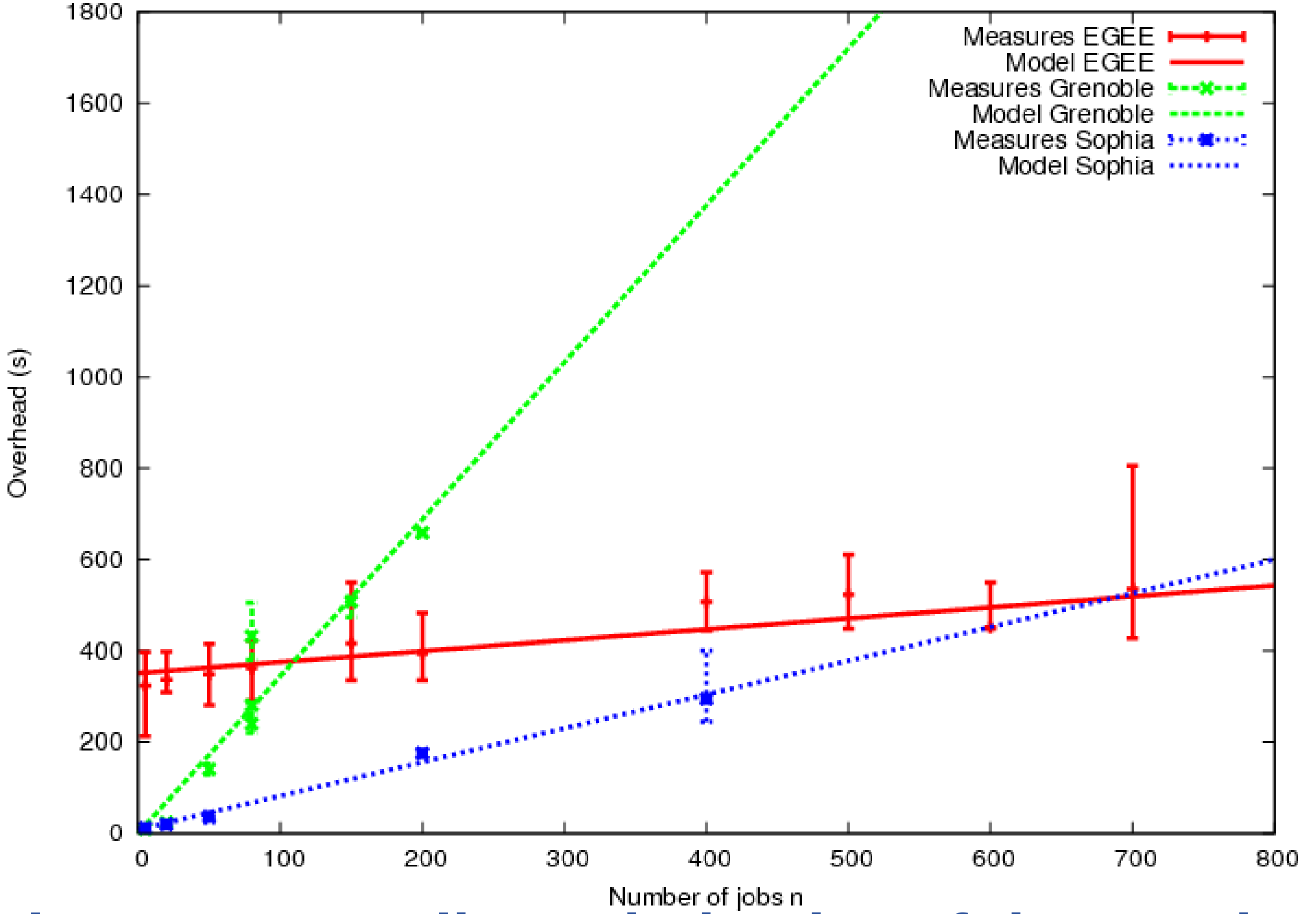
=> Overheads have different orders of magnitude

Measuring the system latency

- **Constant load on the workload management system:**
  - Constant number ***n*** of submitted jobs
  - Resubmission each time a job completed
- **Short jobs:**
  - High turn-over
  - Constant sleeping time t_{run} = 1 minute
- **Measure of the overhead**
  - Measure of t_{exec}-t_{run}
  - Over a 3 hour time-period
  - Increasing number of jobs ***n***

Results

- **Measures on all 3 systems for n increasing**



- **Results suggest a linear behavior of the overhead w.r.t n**
- **Two parameters A (slope) and B (y-intercept)**

Median overhead analysis

- **Metrics**
  - **Nominal overhead**: the y-intercept value **B** of the linear approximation
  - **Scalability**: the inverse of the slope **A** of the linear approximation
- **Results**

Growing size ↓

System	A (s/job)	B (s)
Grid5000 – Grenoble	3.44	0.48
Grid5000 – Sophia	0.74	8.25
EGEE – biomed VO	0.24	351.4

- **Antagonist behaviors of the nominal overhead and scalability**

Application: multi-grids workload

- **The problem: jobs repartition among two systems**
  - ***n*** jobs to submit in parallel
  - 2 systems: median overhead times : $A_1 n+B_1$ and $A_2 n+B_2$
  - $\hat{\delta}$ the optimal fraction of jobs to submit on the first (largest) one
- **$\hat{\delta}$ minimizes the following expression:**

$$H(\delta) = \delta(A_1.\delta.n + B_1) + (1-\delta)(A_2.(1-\delta).n + B_2)$$

- **Optimal proportion of jobs to submit on the largest (1st) system:**

$$\hat{\delta}(n) = \frac{B_2 - B_1 + 2.A_2.n}{2.n.(A_2 + A_1)}$$

$(B_1>B_2\,;\, A_1<A_2)$

  - Transition number $n_{0.5} = \frac{B_1 - B_2}{A_2 - A_1}$
  - Saturation number $\hat{\delta}(\infty) = \frac{A_2}{A_1 + A_2}$

Application to the systems

- **Pairwise comparison:**

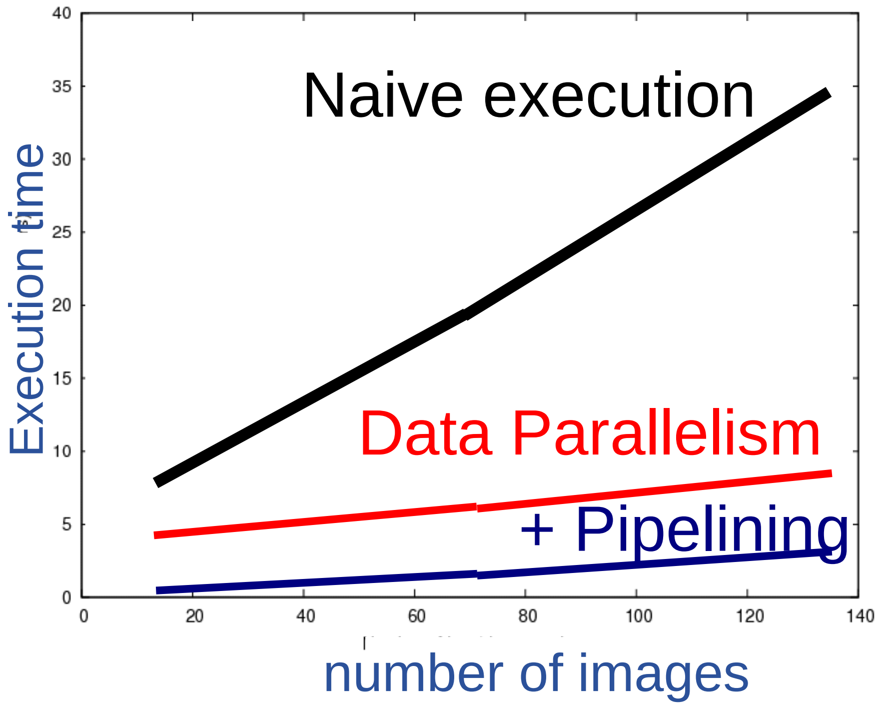
Largest system	Smallest system	n_0	$n_{0.5}$	$\delta(\infty)$
EGEE	Sophia	232 jobs	686 jobs	76%
EGEE	Grenoble	51 jobs	110 jobs	93%
Sophia	Grenoble	1 job	3 jobs	82%

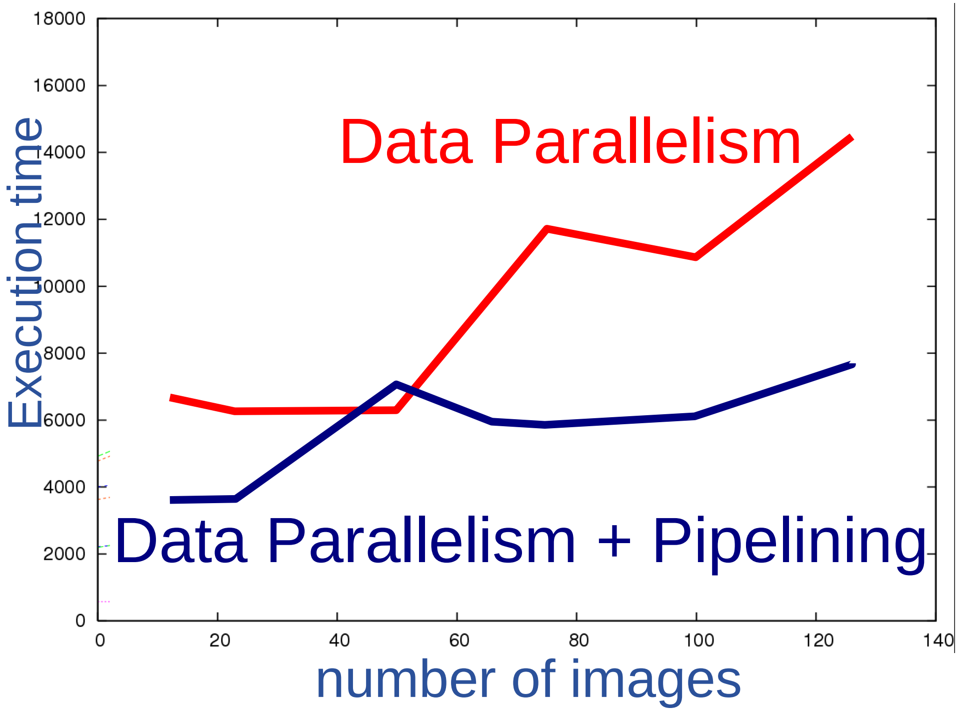
Different load conditions

- **Grid'5000**
  - Reserved resources

EGEE

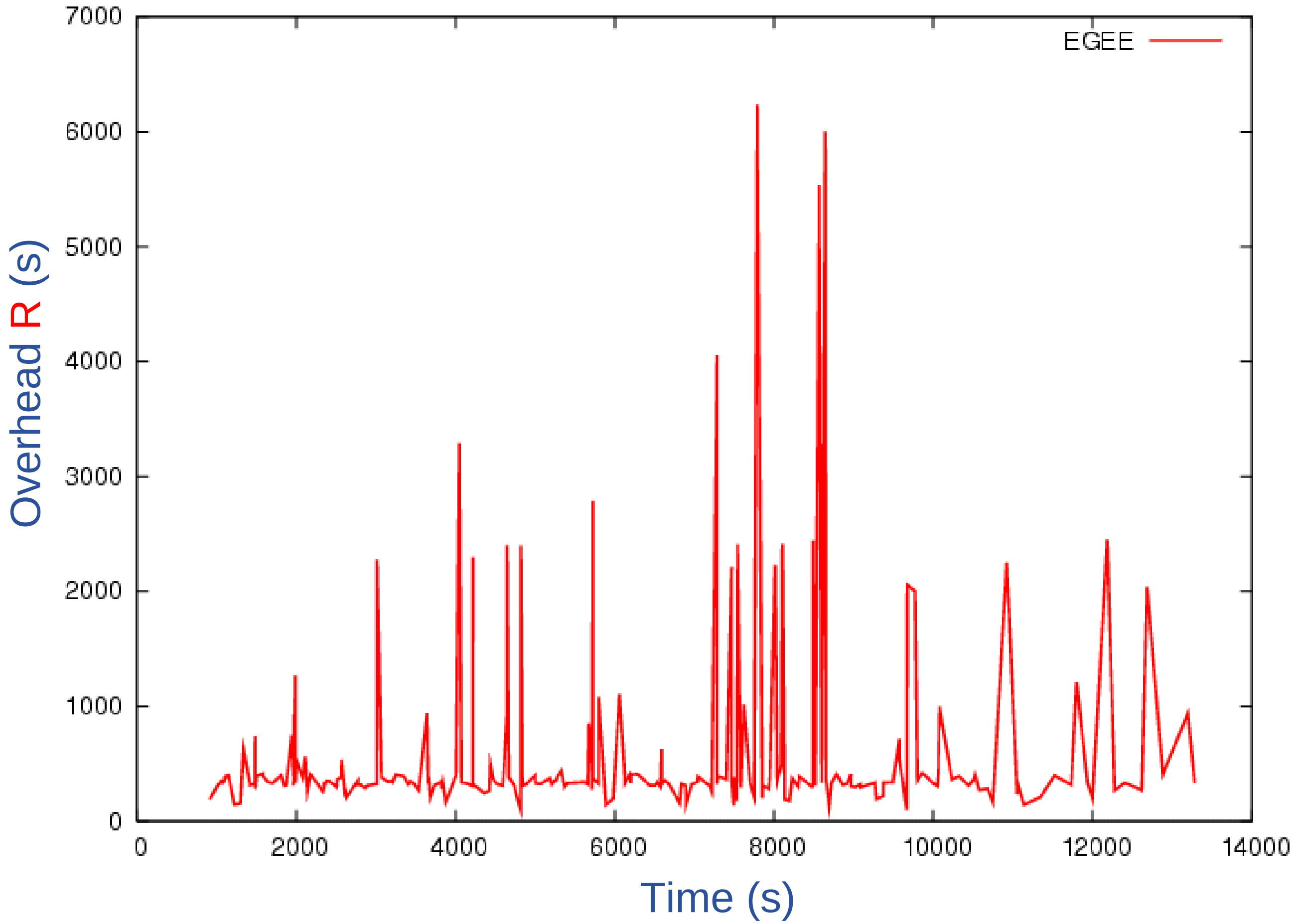
Uncontrolable load conditions





A probabilistic approach

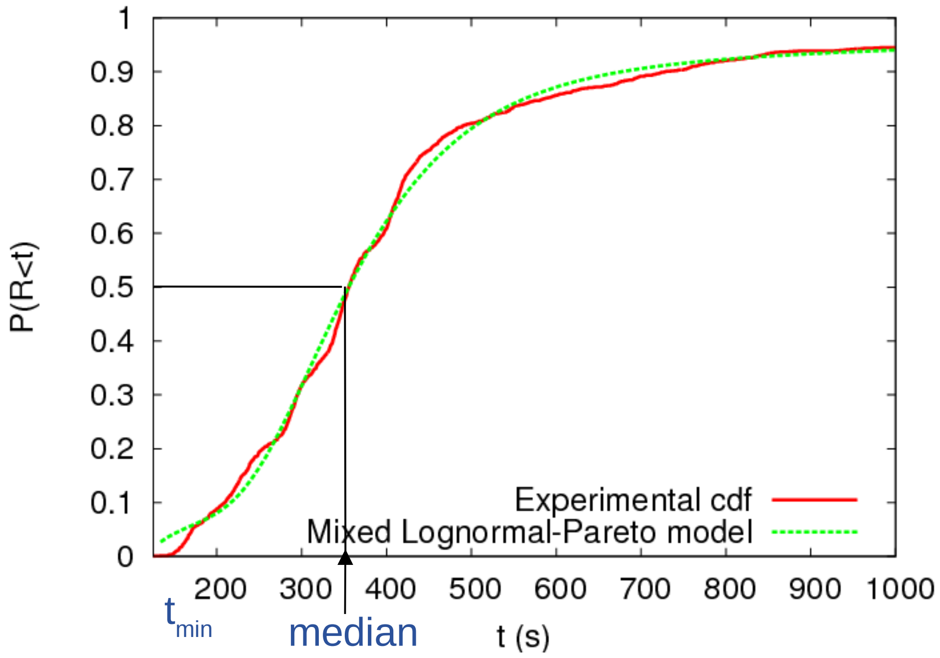
- **Evolution of the overhead time on a 3.5 hour period on EGEE**



- **Let us model R as a random variable**

Model of the latency on EGEE

- **Cumulative density function (c.d.f) of R**
  - 50% of the jobs latency is 350 seconds
  - There are a significant number of jobs with long latency time



- **Data acquisition (red curv**
  - Total: 2137 jobs
  - Outliers threshold: 10.000s
- **Model fitting (green curve)**
  - Heavy tail distributions

$$P^{model}(R<t) = \underbrace{(1-\alpha(t))\,\Phi\left(\frac{\ln(t-t_{\min})-\mu}{\sigma}\right)}_{\text{Body: log-normal}} + \underbrace{\alpha(t)\left(1-\left(\frac{a}{a+t}\right)^{\nu}\right)}_{\text{Tail: Pareto}}$$

Submission optimization

- **Optimize jobs granularity**
  - Trade-off:
  - Minimize number of jobs on high latency systems
  - Maximize parallelism

- **Optimize jobs time-out**
  - Time-out abnormally long jobs (outliers)
  - Compute an estimate to what “abnormally long” mean depending on the infrastructure load

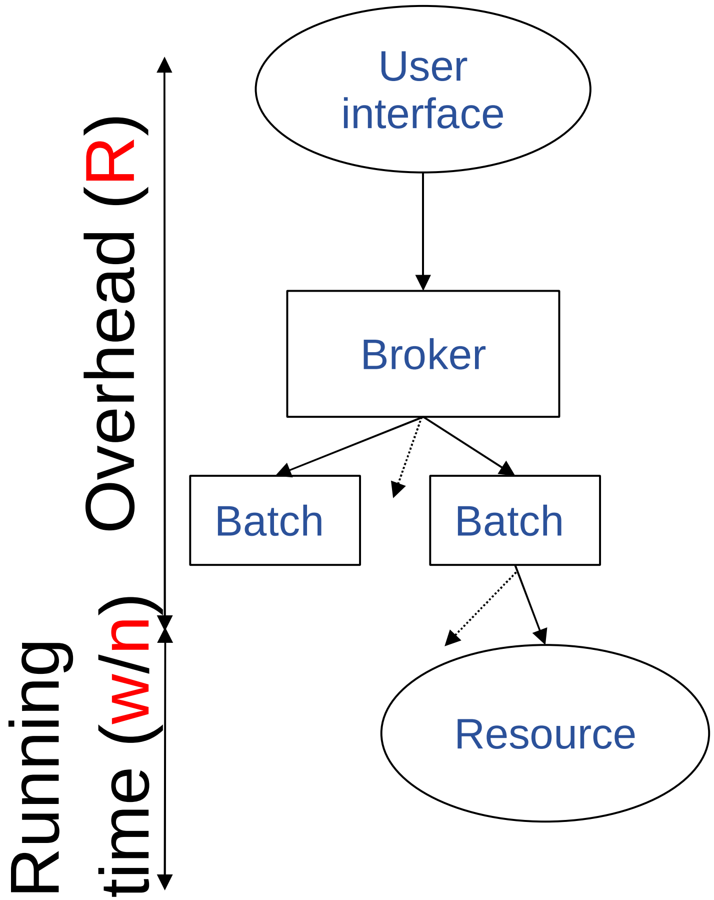
Optimizing granularity

- **Total CPU time of the job to execute: w**
- **Split into n tasks**
- **Random overhead: R**
- **Total execution time H:**

$$H = \max_{i=1..n} (R + w/n)$$

- **Optimum: minimizing H w.r.t n**

$$E_H(n) = \int_{\mathrm{R}} n.t.f_R(t).F_R(t)^{n-1}dt + \frac{W}{n}$$



Timeout optimization

- **Hypotheses**
  - A time-outed job is cancelled then resubmitted
  - Neglect Cancel/Resubmit cost
  - Neglect Cancel/Resubmit overload => independent submissions
- **Execution time from ith submission to completion**

Wall-clock time (r); Latency in normal mode (R); Timeout value (t_∞); Probability to timeout (q)

$$J_i = \begin{cases} r + R & \text{with probability } 1 - q \\ t_\infty + J_{i+1} & \text{with probability } q \end{cases}$$

- **Probability to timeout**

Outlier ratio (ρ)

$$q = \rho + (1-\rho)P(r + R > t_\infty)$$

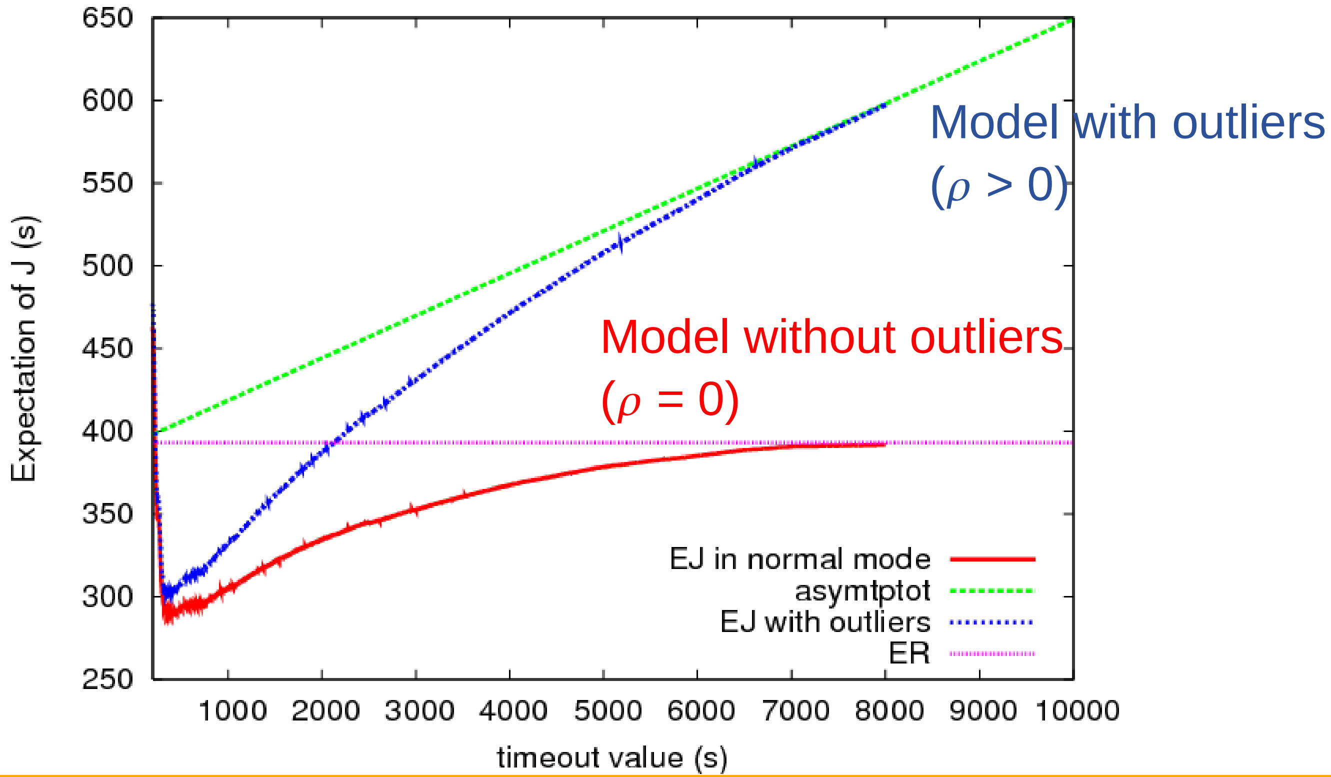
$$q = 1 - (1-\rho)F_R(t_\infty - r)$$

Experiments on EGEE

- **Minimization criterion**

$$E_J(t_\infty) = \frac{1}{F_R(t_\infty)} \int_0^{t_\infty} u f_R(u) du + \frac{t_\infty}{(1-\rho)F_R(t_\infty)} - t_\infty$$

- **Using the distribution measured on EGEE:**

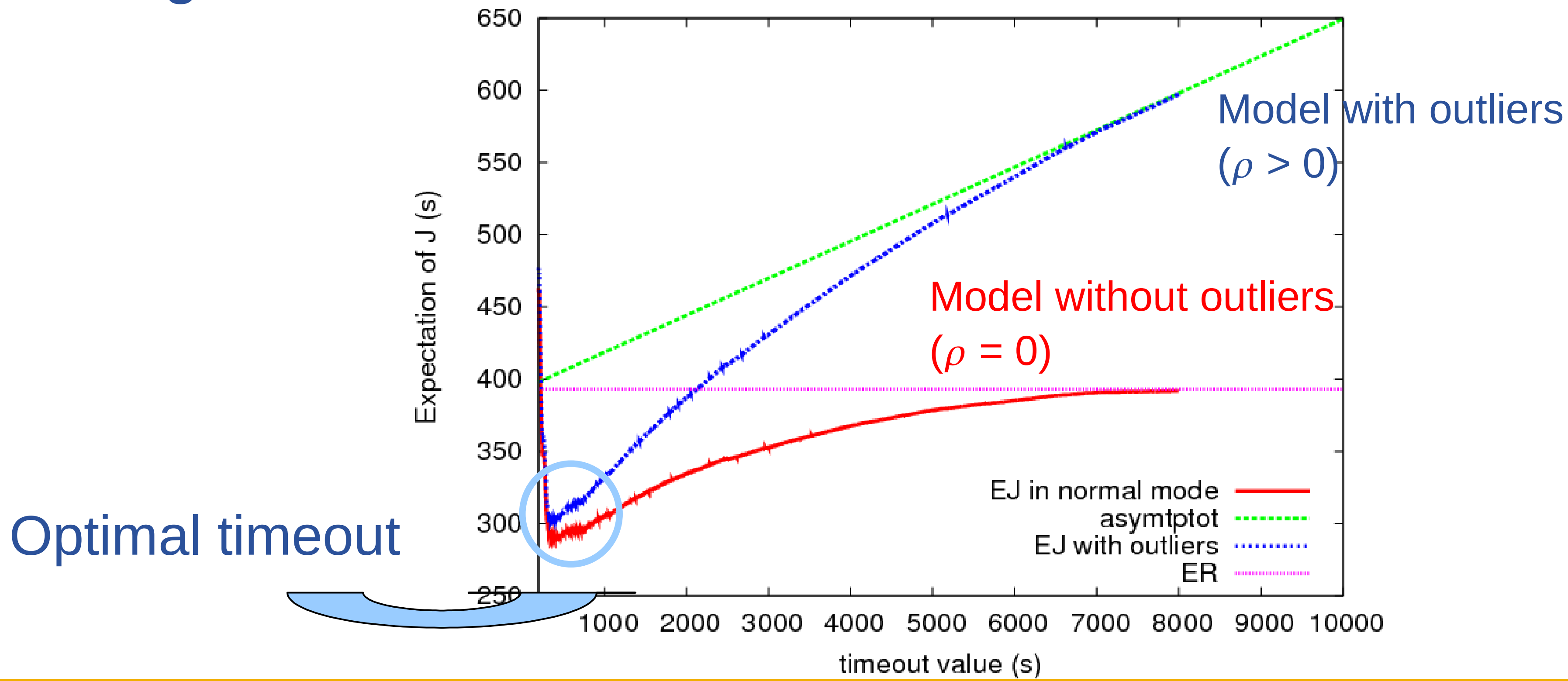


Experiments on EGEE

- **Minimization criterion**

$$E_J(t_\infty) = \frac{1}{F_R(t_\infty)} \int_0^{t_\infty} u f_R(u) du + \frac{t_\infty}{(1-\rho)F_R(t_\infty)} - t_\infty$$

- **Using the distribution measured on EGEE:**



Conclusions and perspectives

- **Overhead monitoring**
  - Directly from EGEE Workload Management System logs
  - To be updated along time
- **Model refinement**
  - More contextual parameters
  - Model validation in controlled environment (e.g. Grid'5000)