Simulation of Price-sensitive Resource Brokering and the Hybrid Pricing Model with DGAS-Sim

Rosario M. Piro
Andrea Guarise, Albert Werbrouck

Istituto Nazionale di Fisica Nucleare (INFN), Turin/Italy
Grid Computing – some difficulties

- The grid is *dynamic* in nature: Resources are added/removed or are not always available (e.g. due to network problems or system failures).
- Resources usually belong to different administrative domains with different policies for access and utilization.
- The utilization of the single resources (and hence their workload) is *unpredictable*:
  - scheduling algorithms usually used for parallel computing (e.g. for cluster computing) are less adequate;
  - grid computing requires more flexible "just-in-time" scheduling for the submitted jobs
Preamble

- research work done within the European DataGrid Project (EDG), that is now being continued as EGEE project (Enabling Grids for E-science in Europe).

- the simulations are based on the architecture of the EDG Workload Management System, adapted to support economic brokering. In the EGEE architecture, resource pricing issues will probably be included in a more complex policy framework.
EDG Workload Management System
WMS (simplified)
Motivation(1): why grid accounting?

- keep track of the resource usage by the single grid users
- avoid abuses of the resources (e.g. due to limited budgets)
- brokering (resource selection) based on economic principles may help the Workload Management System to balance the workload.

facilitate the fair and balanced exchange of computing resources
DataGrid Accounting System (DGAS)
Interaction between DGAS and other EDG components (simplified)
Motivation(2): why economic brokering?

- all grid entities (users, computing resources, storage resources,…) spend or earn "virtual credits" (GridCredits) for their grid related activities.

- resource prices are determined *dynamically*. An *economic feedback* may help the Resource Broker to select less loaded (possibly idle) resources for the submitted jobs, if the prices reflect the resources' workloads.

- allows for a *decentralized* optimization of the distribution of jobs (offering the *scalibility* desired for grid computing).

- may bring about *market equilibrium* in which the requested computing power equals the computing power furnished by the resources.
Hybrid Pricing Model (HPM)

- pricing algorithm (price computed for each single resource):

\[
\text{price} = P_0 + \Delta P \frac{W - \frac{1}{2}W_{\text{max}}}{\frac{1}{2}W_{\text{max}}} \quad 0 \leq W \leq W_{\text{max}}
\]

- static parameters: base price \(P_0\), variation limit \(\Delta P\) and maximum Queue Wait Time \(W_{\text{max}}\)

- dynamic parameter: current Queue Wait Time \(W\) (estimated)

Price dynamically varies between \(P_0 - \Delta P\) (empty queue) and \(P_0 + \Delta P\) (full queue) as a function of the resource's Queue Wait Time.
Computational Energy

- Prices are expressed in Grid Credits per Unit of Computational Energy (GC/UCE)
- We define the computational energy "consumed" by a job as the product of a performance factor or power $p$ (e.g. benchmark for processing power) and the amount of usage $u$ (e.g. processing time).
- The consumption of computational energy by a job is ideally independent of the resource that is executing it.
- For the presented simulation:
  - assume the clock speed to be a sufficiently good measure for processing power
  - $1 \text{ UCE} = 1 \text{ MHz} \times \text{ min.}$

WETICE (ETNGRID) 2004, Modena, Italy

piro@to.infn.it
Advantages and disadvantages of the HPM

- simplicity, low computational overhead and low communication overhead (only minimal information required).
- variation limits prevent the degeneration of resource prices (e.g. due to inflation).
- requires definition of maximum QWTs for predefined queue classes (e.g. short, medium, long ..)
- requires a reasonable estimate of current QWTs (and thus an estimation of job requirements in terms of computational energy).

=> possible: estimation by user or based on historical data for similar jobs (W.Smith, V.Taylor and I.Foster, 1999; H.Li, D.Groep, J.Templon and L.Wolters, 2004).
DGAS-Sim(ulator)

- modular and expandable tool to *simulate the components involved in the distribution of the workload* (Resource Broker, Grid Information System, Price Authorities, Computing Elements).
- interfaces for brokering strategies and pricing algorithms allow the simulation of different behaviors of Resource Brokers and Price Authorities to *study their impact on workload balancing*.
- a *configuration file* allows to specify an arbitrary number of Computing Elements (resources) and Price Authorities, the interval between price adjustments, different job types and their consumption of computational energy and their requirements in terms of computing power, etc.
- simulation of *single-processor resources with one FCFS queue* (we focus on load balancing *among* grid resources).
- data location and network traffic are not considered.
DGAS-Sim: architecture

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```

```
DGAS-Sim: architeture
```
Brokering strategies

- **price-sensitive**: prices are the only criterion for resource selection; the Resource Broker chooses the resource (Computing Element) with the *lowest price*; other factors such as data access times and network bandwidths are not considered.

For comparison:

- **random** (no optimization): jobs are submitted to *randomly chosen* resources that satisfy their requirements.
- **completion time** (best case): the criterion for resource selection is the given job's *completion time* (QWT + execution time); the Resource Broker chooses the resource with the *lowest completion time* (that is assumed to be exactly known).
First simulation results (I)

- pricing algorithm: HPM with small price variations (about ±5%); price adjustment interval of about 12 seconds; small number of resources
- brokering strategy: price-sensitive
- the queues of the resources are *initially unbalanced*
- three cases:
  - market equilibrium (demand = supply)
  - overdemand (demand > supply)
  - underdemand (demand < supply)
First simulation results (II)

- **Demand = Supply**
  - Graph showing QWT of HPM at market equilibrium; equal base price; equal CPU power.
  -Submitted jobs required 123.5 UCE/t.u. (mean value).
  -Maximum QWT is 2000 time units (1000 s).

- **Demand < Supply**
  -Graph showing QWT of HPM, under-demand case; equal base price; equal CPU power.
  -Submitted jobs required 96.5 UCE/t.u. (mean value).
  -Maximum QWT is 2000 time units (1000 s).

- **Demand > Supply**
  -Graph showing QWT of HPM, over-demand case; equal base price; equal CPU power.
  -Submitted jobs required 178.5 UCE/t.u. (mean value).
  -Maximum QWT is 2000 time units (1000 s).
Simulation of a small grid

- 50 Computing Elements (CEs)
  - with **5 different queue types**:
    - *very short queues*: maximum QWT of 2 hours
    - *short queues*: maximum QWT of 4 hours
    - *medium queues*: maximum QWT of 10 hours
    - *long queues*: maximum QWT of 18 hours
    - *very long queues*: maximum QWT of 24 hours
  - with **5 different processing powers**:
    - 2 CEs of each queue type with 800 UCE/min (~ 800 MHz)
    - 2 CEs of each queue type with 1200 UCE/min (~ 1.2 GHz)
    - 2 CEs of each queue type with 1600 UCE/min (~ 1.6 GHz)
    - 2 CEs of each queue type with 2000 UCE/min (~ 2.0 GHz)
    - 2 CEs of each queue type with 2400 UCE/min (~ 2.4 GHz)
Simulation of a small grid

- one job submission about every 30 seconds; 20 different job types with different requirements in terms of processing power and queue types (to simulate matchmaking):
  - *underdemand* for shorter queues,
  - *overdemand* for longer queues.
- brokering strategies: price-sensitive, random, completion time
- pricing algorithm (for price-sensitive brokering): HPM with a price adjustment interval of about 10 minutes and *initially unbalanced* queues
  - *Sim. run 1*: base prices $P_0$ between 775 and 1225 GC/UCE; *low price variation limit* $\Delta P$ of 50 GC/UCE (=> price domains do not overlap significantly)
  - *Sim. run 2*: *common base price* of $P_0 = 1000$ GC/UCE for all CEs; price variation limit $\Delta P$ of 50 GC/UCE
Results (1): QWT, very short queues

QWT of very short queues (120 time units = 2 hrs); Random

Computing energy furnished:
- CE0 and CE1: 800 UCE/t.u.
- CE2 and CE3: 1200 UCE/t.u.
- CE4 and CE5: 1600 UCE/t.u.
- CE6 and CE7: 2000 UCE/t.u.
- CE8 and CE9: 2400 UCE/t.u.
Utility function: random

QWT of very short queues (120 time units = 2 hrs); min. Completion Time

Computing energy furnished:
- CE0 and CE1: 800 UCE/t.u.
- CE2 and CE3: 1200 UCE/t.u.
- CE4 and CE5: 1600 UCE/t.u.
- CE6 and CE7: 2000 UCE/t.u.
- CE8 and CE9: 2400 UCE/t.u.
Utility function: min. completion time

QWT of HPM for very short queues (120 time units = 2 hrs)

Computing energy furnished:
- CE0 and CE1: 800 UCE/t.u.
- CE2 and CE3: 1200 UCE/t.u.
- CE4 and CE5: 1600 UCE/t.u.
- CE6 and CE7: 2000 UCE/t.u.
- CE8 and CE9: 2400 UCE/t.u.

Prices of HPM:
- CE0-2: price is 1175 +/- 50
- CE3-6: price is 1200 +/- 50
- CE7-9: price is 1225 +/- 50

price adjustment interval: 10min,
price adjustment threshold is 2
Utility function: price-sensitive

QWT of HPM for very short queues (120 time units = 2 hrs)

Computing energy furnished:
- CE0 and CE1: 800 UCE/t.u.
- CE2 and CE3: 1200 UCE/t.u.
- CE4 and CE5: 1600 UCE/t.u.
- CE6 and CE7: 2000 UCE/t.u.
- CE8 and CE9: 2400 UCE/t.u.

Base prices of HPM:
- CE0-9: base price is 1000
price variation limits: +/- 50
price adjustment threshold is 2
Utility function: price-sensitive

random

completion time

HPM, different $P_0$, low $\Delta P$
Results (2): QWT, short queues

**QWT of short queues (240 time units = 4 hrs): Random**

- Computing energy furnished:
  - CE10 and CE11: 800 UCE/t.u.
  - CE12 and CE13: 1200 UCE/t.u.
  - CE14 and CE15: 1600 UCE/t.u.
  - CE16 and CE17: 2000 UCE/t.u.
  - CE18 and CE19: 2400 UCE/t.u.
- Utility function: random

**QWT of short queues (240 time units = 4 hrs): min. Completion Time**

- Computing energy furnished:
  - CE10 and CE11: 800 UCE/t.u.
  - CE12 and CE13: 1200 UCE/t.u.
  - CE14 and CE15: 1600 UCE/t.u.
  - CE16 and CE17: 2000 UCE/t.u.
  - CE18 and CE19: 2400 UCE/t.u.
- Utility function: min. completion time

---

**QWT of HPM for short queues (240 time units = 4 hrs): HPM, different P₀, low ΔP**

- Prices of HPM:
  - CE10-12: price is 1075 +/- 50
  - CE13-16: price is 1100 +/- 50
  - CE17-19: price is 1125 +/- 50
- price adjustment interval: 10 min.
- price adjustment threshold is 2
- Utility function: price-sensitive

---

**QWT of HPM for short queues (240 time units = 4 hrs): HPM, common P₀**

- Base prices of HPM:
  - CE10-19: base price is 1000
- price variation limits: +/- 50
- price adjustment threshold is 2
- Utility function: price-sensitive
Results (3): QWT, medium queues

QWT of medium queues (600 time units = 10 hrs); Random

- Computing energy furnished:
  - CE20 and CE21: 800 UCE/t.u.
  - CE22 and CE23: 1200 UCE/t.u.
  - CE24 and CE25: 1600 UCE/t.u.
  - CE28 and CE29: 2400 UCE/t.u.
- Utility function: random

Random completion time

- QWT

QWT of medium queues (600 time units = 10 hrs); min. Completion Time

- Computing energy furnished:
  - CE20 and CE21: 800 UCE/t.u.
  - CE22 and CE23: 1200 UCE/t.u.
  - CE24 and CE25: 1600 UCE/t.u.
  - CE28 and CE29: 2400 UCE/t.u.
- Utility function: min. completion time

HPM, different $P^0$

- Prices of HPM:
  - CE20-22: price is 975 +/- 50
  - CE23-29: price is 1025 +/- 50
- Price adjustment interval: 10 min
- Price adjustment threshold is 2
- Utility function: price-sensitive

HPM, common $P^0$

- Base prices of HPM:
  - CE20-29: base price is 1000
  - Price variation limits: +/- 50
  - Price adjustment threshold is 2
- Utility function: price-sensitive
Results (4): QWT, long queues

- **QWT**
  - QWT of long queues (1080 time units = 18 hrs); Random
  - Computing energy furnished:
    - CE30 and CE31: 800 UCE/l.u.
    - CE32 and CE33: 1200 UCE/l.u.
    - CE34 and CE35: 1600 UCE/l.u.
    - CE36 and CE37: 2000 UCE/l.u.
    - CE38 and CE39: 2400 UCE/l.u.
  - Utility function: random

- **QWT**
  - QWT of HPM for long queues (1080 time units = 18 hrs)
  - Computing energy furnished:
    - CE30 and CE31: 800 UCE/l.u.
    - CE32 and CE33: 1200 UCE/l.u.
    - CE34 and CE35: 1600 UCE/l.u.
    - CE36 and CE37: 2000 UCE/l.u.
    - CE38 and CE39: 2400 UCE/l.u.
  - Prices of HPM:
    - CE30-32: price is 875 +/- 50
    - CE33-36: price is 900 +/- 50
    - CE37-39: price is 925 +/- 50
  - Price adjustment interval: 10min.
  - Price adjustment threshold is 2
  - Utility function: price-sensitive

- **QWT**
  - QWT of long queues (1080 time units = 18 hrs); min. Completion Time
  - Computing energy furnished:
    - CE30 and CE31: 800 UCE/l.u.
    - CE32 and CE33: 1200 UCE/l.u.
    - CE34 and CE35: 1600 UCE/l.u.
    - CE36 and CE37: 2000 UCE/l.u.
    - CE38 and CE39: 2400 UCE/l.u.
  - Utility function: min. completion time

- **QWT**
  - QWT of HPM for long queues (1080 time units = 18 hrs)
  - Computing energy furnished:
    - CE30 and CE31: 800 UCE/l.u.
    - CE32 and CE33: 1200 UCE/l.u.
    - CE34 and CE35: 1600 UCE/l.u.
    - CE36 and CE37: 2000 UCE/l.u.
    - CE38 and CE39: 2400 UCE/l.u.
  - Base prices of HPM:
    - CE30-39: base price is 1000
  - Price variation limits: +/- 50
  - Price adjustment threshold is 2
  - Utility function: price-sensitive

**Legend:**
- **Random** completion time
- **Low ΔP**
- **HPM, different P₀**
- **HPM, common P₀**
Results (5): QWT, very long queues

QWT of very long queues (1440 time units = 24 hrs); Random

- Computing energy furnished:
  - CE40 and CE41: 800 UCE/lu.
  - CE42 and CE43: 1200 UCE/lu.
  - CE44 and CE45: 1600 UCE/lu.
  - CE46 and CE47: 2000 UCE/lu.
  - CE48 and CE49: 2400 UCE/lu.
- Utility function: random

QWT of very long queues (1440 time units = 24 hrs); min. Completion Time

- Computing energy furnished:
  - CE40 and CE41: 800 UCE/lu.
  - CE42 and CE43: 1200 UCE/lu.
  - CE44 and CE45: 1600 UCE/lu.
  - CE46 and CE47: 2000 UCE/lu.
  - CE48 and CE49: 2400 UCE/lu.
- Utility function: min. completion time

QWT of HPM for very long queues (1440 time units = 24 hrs)

- HPM, different $P_0$
- low $\Delta P$

=- Computing energy furnished:
  - CE40 and CE41: 800 UCE/lu.
  - CE42 and CE43: 1200 UCE/lu.
  - CE44 and CE45: 1600 UCE/lu.
  - CE46 and CE47: 2000 UCE/lu.
  - CE48 and CE49: 2400 UCE/lu.
- Prices of HPM:
  - CE40-42: price is 775 +/- 50
  - CE43-46: price is 800 +/- 50
  - CE47-49: price is 825 +/- 50
- Price adjustment interval: 10min.
- Price adjustment threshold is 2
- Utility function: price-sensitive

QWT of HPM for very long queues (1440 time units = 24 hrs)

- HPM, common $P_0$

- Utility function: price-sensitive
Results (6): relative standard deviation

QWT

relative standard deviation of QWT of HPM

Utility function: random

QWT

relative standard deviation of QWT of HPM

Utility function: min. completion time

QWT

relative standard deviation of QWT of HPM

HPM, different $P_0$

low $\Delta P$

Price of HPM:
- medium queues: 975,1000,1025 (+/-50)
- long queues: 875,900,925 (+/-50)
- very long queues: 775,800,825 (+/-50)

price adjustment interval: 10min,
price adjustment threshold is 2
Utility function: price-sensitive

QWT

relative standard deviation of QWT of HPM

HPM, common $P_0$

Prices of HPM:
- all queues: 1000 (+/-250)

price adjustment interval: 10min,
price adjustment threshold is 2
Utility function: price-sensitive
Importance of overlapping price domains

- if using price-sensitive brokering an effective load balancing requires widely overlapping price domains:
  - *Sim. run 2*: common base price of $P_0 = 1000$ GC/UCE for all CEs; price variation limit $\Delta P$ of 50 GC/UCE
  - *Sim. run 3*: base prices $P_0$ between 775 and 1225 GC/UCE (the same base prices as in sim. run 1); high price variation limit $\Delta P$ of 250 GC/UCE (=> price domains are widely overlapping!)

- non-overlapping price domains lead to "subgrids" with eventually different levels of workload (which may also be a desirable setting, where users pay more for immediate processing, e.g. for interactive jobs)
Results (7): QWT, very short & short queues

**QWT**

- **HPM, common P**
  - Very short queues
  - HPM, different P
    - High ∆P

- **QWT**
  - Computing energy furnished:
    - CE0 and CE1: 600 UCE/t.u.
    - CE2 and CE3: 1200 UCE/t.u.
    - CE4 and CE5: 1600 UCE/t.u.
    - CE6 and CE7: 2000 UCE/t.u.
    - CE8 and CE9: 2400 UCE/t.u.

- **QWT**
  - Computing energy furnished:
    - CE10 and CE11: 800 UCE/t.u.
    - CE12 and CE13: 1200 UCE/t.u.
    - CE14 and CE15: 1600 UCE/t.u.
    - CE16 and CE17: 2000 UCE/t.u.
    - CE18 and CE19: 2400 UCE/t.u.

- **Base prices of HPM:**
  - CE0-9: base price is 1000
  - Price variation limits: +/- 50
  - Price adjustment threshold is 2
  - Utility function: price-sensitive

- **QWT**
  - Computing energy furnished:
    - CE10 and CE11: 800 UCE/t.u.
    - CE12 and CE13: 1200 UCE/t.u.
    - CE14 and CE15: 1600 UCE/t.u.
    - CE16 and CE17: 2000 UCE/t.u.
    - CE18 and CE19: 2400 UCE/t.u.

- **Base prices of HPM:**
  - CE10-19: base price is 1000
  - Price variation limits: +/- 50
  - Price adjustment threshold is 2
  - Utility function: price-sensitive
Results (8): QWT, medium & long queues

QWT of HPM for medium queues (600 time units = 10 hrs)

HPM, common $P_0$

QWT of HPM for long queues (1080 time units = 18 hrs)

HPM, common $P_0$

QWT of HPM for medium queues (600 time units = 10 hrs)

HPM, different $P_0$, high $\Delta P$

QWT of HPM for long queues (1080 time units = 18 hrs)

HPM, different $P_0$, high $\Delta P$
Results (9): QWT, very long q. & rel. std. dev.

**QWT of HPM for very long queues (1440 time units = 24 hrs)**

- **HPM, common** $P_0$
- **very long queues**

**QWT of HPM for high $\Delta P$**

- **HPM, different** $P_0$
- **high $\Delta P$**

Base prices of HPM:
- CE40 and CE41: 800 UCE/t.u.
- CE42 and CE43: 1200 UCE/t.u.
- CE44 and CE45: 1600 UCE/t.u.
- CE48 and CE49: 2400 UCE/t.u.

Computing energy furnished:
- CE40 and CE41: 800 UCE/t.u.
- CE42 and CE43: 1200 UCE/t.u.
- CE44 and CE45: 1600 UCE/t.u.
- CE48 and CE49: 2400 UCE/t.u.

Relative standard deviation of QWT of HPM:

- medium queues
- long queues
- very long queues

**Prices of HPM:**
- medium queues: 975, 1000, 1025 (±250)
- long queues: 875, 900, 925 (±250)
- very long queues: 775, 800, 825 (±250)

Price adjustment interval: 10 min.
Price adjustment threshold is 2
Utility function: price-sensitive
Conclusions

- DGAS-Sim allows to study the impact of different brokering strategies and pricing algorithms on load balancing in computational grids before deploying them on testbeds or production grids.

- Most previous simulations of computational economies for grid computing focused on reaching a market equilibrium by computing an equilibrium price valid for all transactions, without considering the workload on the single resources.

- The Hybrid Pricing Model – provided an appropriate configuration – approximates the "best case" (completion time-based brokering), additionally offering the advantages of an economic approach:
  - Possibility to reach market equilibrium (requested computational energy equals furnished computational energy) by incentivating users to delay less urgent jobs to periods with lower congestion.
  - An economic framework may include data management, data storage, networking and priority (limited budgets) issues.