Delay Components of Job Processing in a Grid: Statistical Analysis and Modeling

K. Christodoulopoulos, V. Gkamas, E. A. Varvarigos
Computer Engineering and Informatics Dept. and
Research Academic Computer Technology Institute,
University of Patras, Patras, Greece
{kchristodou, gkamas, manos}@ceid.upatras.gr

Abstract

The existence of good probabilistic models for the job arrival process and the delay components introduced at the different stages of job processing in a Grid environment is important for the improved understanding of the computing concept envisioned by the Grid. In this study, we present a thorough analysis of the job arrival process in the EGEE infrastructure and the time durations a job spends at different states in the EGEE environment. We define four delay components of the total job delay and model each component separately. We observe that the job inter-arrival times at the Grid level can be adequately modeled by a rounded exponential distribution, while the total job delay (from the time it is generated until the time it completes execution) is dominated by the Computing Element's queuing and Worker Node's execution times.

1. Introduction

Grids introduce new ways to share computing and storage resources across geographically separated sites by establishing a global resource management architecture [1]. The job inter-arrival times, the job execution times, and the times jobs spent at different phases of processing in Grids are unknown and are better modeled probabilistically. Finding good probabilistic models for the job submission process, the job delay components, and the job characteristics is important for the improved understanding of grid systems. Such models would facilitate the dimensioning of grid systems, the prediction of their performance, the improvement of the middleware they use, and the evaluation of new scheduling and quality of service policies.

Even though a large number of works on job characterization and modeling for single parallel supercomputers have appeared in the literature [9], [10], the corresponding attempts in the area of Grid computing are quite limited [4], [5]. In [4], Medernach analyzed and modeled the workload of a LCG/EGEE cluster. More specifically, a two-dimensional Markov chain, which is equivalent to a 2-phase hyper exponential process, was proposed for modeling user behavior in a Grid environment. The user shifts between login and logout states and submits jobs when being in the login state. The results indicate that the 2-phase hyper exponential process can satisfactorily model the submission behavior of a single user.

Taking a different approach, Li et al [5] used the LCG Real Time Monitor tool [3] in order to collect data from Resource Brokers (RBs) located at CERN, Germany, and the UK, and proposed traffic models for the job arrival processes at three different levels: Grids, Virtual Organizations and regions. By comparing a set of m-state Markov modulated Poisson processes (MMPP) with Poisson and hyper exponential processes, they conclude that MMPP models with a certain number of states are capable of modeling the submitted job traffic at the three examined levels. However, the proposed models are not intuitive enough, and they do not provide an easy, adaptable or extensible way for profiling arrival processes in Grid environments.

The measurements that are presented in the present study correspond to the Grid level, meaning that we have considered the overall LCG/EGEE infrastructure as a single entity and observed the general properties of job submission and execution in this real and highly-utilized Grid environment. Based on the LCG/EGEE job flow diagram, we distinguish four delay components of the job processing, each corresponding to time spent at different states in the LCG/EGEE environment, from the submission of a job until the retrieval of the corresponding output data. We then analyze and model each delay component separately.

Our results indicate that if we consider the LCG/EGEE Grid as the level of our observation, the job interarrival times were found to match very well with a rounded exponential distribution with mean 1.6077 sec. We then proceeded to define and model the four delay components that comprise the overall job processing in the LCG/EGEE environment. More specifically, the first delay component (D_1) corresponds to the time a job spends in the Pending, Submitted and Waiting states and can be adequately modeled as a deterministic (constant) parameter. The second delay component (D_2) corresponds

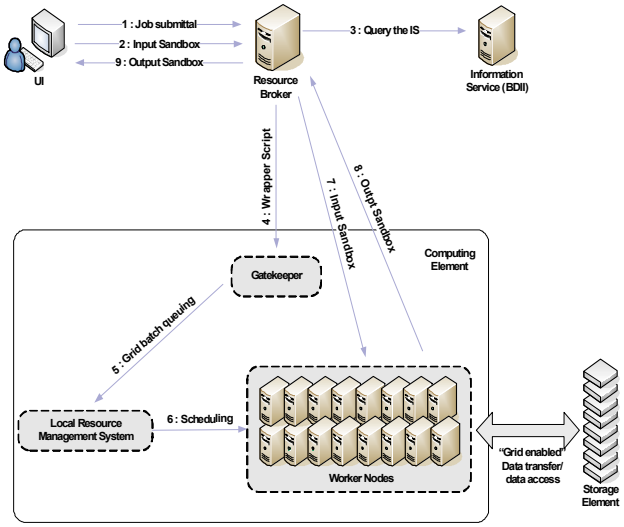

Figure 1 - Job flow in the LCG/EGEE environment

to the time a job stays in the Ready state and can be modeled very well by a 2 phase lognormal distribution, and less accurately by the sum of a deterministic and a lognormal distribution or a 3 phase hyper-exponential distribution. We observed that the total time a job stays in the LCG/EGEE environment is dominated by the Computing Element's Queuing delay and the Worker Node's (WN) execution time that correspond to the third (D_3) and fourth (D_4) delay components, respectively. Similar to D_2, the queuing times (D_3) can be modeled quite accurately by a 2 phase lognormal distribution, and less accurately by the sum of a deterministic and a lognormal variable or a 3 phase hyper exponential distribution. For the WN execution time D_4, we found that a hyper-exponential process with 3 states is sufficient for modeling the stepwise patterns observed in the empirical distribution obtained from our measurements.

The rest of this work is organized as follows. The LCG/EGEE environment is presented in Section 2. Section 3 describes the job flow in the LCG/EGEE environment and the various metrics used for the analysis of the job arrival process and the job delay components. Section 4 presents the statistical results obtained. The modeling of the interarrival times and the job processing delay components is presented in Section 5. Section 6 concludes the study.

2. LCG/EGEE Projects And Infrastructure

The Enabling Grids for E-sciencE (EGEE) project [2] aims at providing researchers with access to a geographically distributed Grid infrastructure, available 24 hours a day. It focuses on maintaining the gLite middleware [6] and on operating the infrastructure for the benefit of a large and diverse research community.

The World wide LHC Computing Grid Project (LCG) [6] was created to prepare the computing infrastructure for the simulation, processing and analysis of the data of the Large Hadron Collider (LHC) experiments. The LCG and the EGEE projects share a large part of their infrastructure and operate it in conjunction. For this reason, we will refer to it as the LCG/EGEE infrastructure. Currently, 207 clusters (sites) from 48 different countries participate in the LCG/EGEE infrastructure [7]. In the observation period of this study, there were totally 39697 CPUs and about 5 Petabytes of storage, while the total average number of available CPUs was 31228 [8].

In the LCG/EGEE environment, users are organized in Virtual Organizations (VOs), which are dynamic collections of individuals and institutions sharing resources in a flexible, secure and coordinated manner. A user has to belong to a VO to be able to use the LCG/EGEE infrastructure. A list of existing VOs in the EGEE is available at [11].

3. Job Flow in the LCG and Used Metrics

Generally, a user cannot submit a job directly to a cluster (site), but he first has to login to a local User Interface (UI). The description of the job is written in a specific format (JDL – job description language). This is forwarded to the corresponding Resource Broker (RB) where the matching process is performed [6]. An RB runs the Workload Management System (WMS) service that intercommunicates with the Information System (IS, providing information about the Grid resources and their status). The RB uses the job description, the related VO and available global load information to decide about whether or not and where to forward the job. Users, when submitting a job, give a rough estimate of its maximum running time, but this value is usually overestimated and is considerably larger than the actual job execution time.

When a job is submitted to the LCG/EGEE environment it passes through several states till the user gets back the desired output data. These states insert corresponding delay components to the total job processing time. The job flow from its submission from a UI, till the retrieval of the job output is shown in Figure 1 [4]. Figure 2 presents the various states of a job in the LCG/EGGE environment. These states come from the gLite 3 user's guide [6] enhanced with a new state (Pending state) and specific time instances (Epochs) useful for the analysis of the interarrival times and the delay components that comprise the job execution in the LCG/EGEE environment.

The time instances (Epochs) of specific events of interest to us for the purposes of modeling are the following:

- **V_1 = userinterface_regjob_Epoch**: The time instance the user submits a job from the UI to a Resource Broker.
- **V_2 = networkserver_accepted_Epoch**: The time instance the job is accepted by the Network Server of the Resource Broker.
- **V_3 = workloadmanager_match_Epoch**: The time instance the WMS starts looking for the best available CE to execute the job.
- **V_4 = jobcontroller_transfer_Epoch**: The time instance the job controller of the RB starts sending the job request to the appropriate CE.
- **V_5 = logmonitor_accepted_Epoch**: The time instance the CE receives the request.
- **V_6 = lrms_running_Epoch**: The time instance the Loc-al Resource Management System (LRMS) assigns the job for execution to an available WN in the local farm.
- **V_7 = logmonitor_running_Epoch**: The time instance the user files have completed transferring from the RB to the WN where the job will be executed.
- **V_8 = lrms_done_Epoch**: The time instance the CE starts transferring the output back to the RB node.
- **V_9 = logmonitor_done_Epoch**: The time instance after which the user can retrieve the job output to the UI.

Based on the aforementioned Epochs, we can define the various states (Figure 2) at which a job can be at any given time in the LCG/EGGE environment:

- The status of the job becomes **Pending** at the time instance V_1 (userinterface_regjob_Epoch) at which the job (more specifically, the job JDL file) is submitted from the UI to the RB.
- The RB receives the JDL file, which may specify one or more files to be copied from the UI to the Worker Node. This set of files is referred to as the Input Sandbox. The status of the job becomes **Submitted** at the time instance V_2 (networkserver_accepted_Epoch) at which the Network Server of the RB accepts the job.
- The RB node runs the WMS service whose role is to find the best available CE to execute the job based on the requirements the user has specified in the JDL file and the state of every site. The WMS service starts to execute at time V_3 (workloadmanager_match_Epoch) at which point the status of the job becomes **Waiting**.
- The RB creates a wrapper script to be passed, together with other parameters, to the selected CE. The status of job becomes **Ready** at the time instance V_4 (jobcontroller_transfer_Epoch) at which the job controller of the RB sends the job to the appropriate CE.

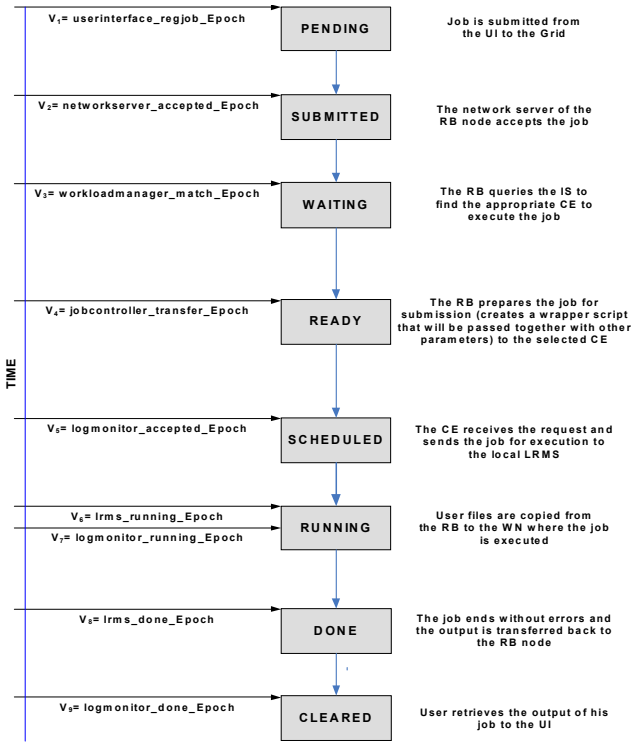

Figure 2 - The states of a job in the LCG/EGEE environment and the corresponding time instances (Epochs)

- The CE receives the request at the time instance V_5 (logmonitor_accepted_Epoch) and the Gatekeeper of the CE sends the job for execution to the LRMS. The status of the job then becomes **Scheduled**.
- The LRMS is the service running at the CE and is responsible for the handling of the job execution on the local farm of WNs. A job remains in the LRMS queue until the time instance V_6 (lrms_running_Epoch) at which the LRMS assigns the job to a WN, at which time the status of the job becomes **Running**. The user files complete transferring from the RB to the WN at time V_7 (logmonitor_running_Epoch).
- If the job completes without errors, the output of the job (called Output Sandbox), starts transferring back to the RB node at time instance V_8 (lrms_done_Epoch), at which point the status of the job becomes **Done**.
- At time instance V_9 (logmonitor_done_Epoch) the Output Sandbox has completed transferring and the user can retrieve the output of his job to the UI. The status of the job becomes and remains **Cleared**.

Using the previous Epochs we calculate the metrics shown in Table 1. These metrics will be used for the analysis of the various delay components that comprise the job execution in the LCG/EGEE environment.

Table 1: Metrics Used For Analysis Of The Various States Of The Job In The LCG/EGEE Environment

Variables	Corresponding States	
V_{10}= registration_Time	Pending	(V_2-V_1)
V_{11}= match_Time	Submitted	(V_3-V_2)
V_{12}=ready_to_transfer_to_CE_T	Pending+Submitted+ Waiting	(V_4-V_1)
V_{13}= transfer_Time	Ready	(V_5-V_4)
V_{14}=logmonitor_CE_total_Time	Scheduled+Running+ Done	(V_9-V_5)
V_{15}= logmonitor_CE_queue_T	Scheduled	(V_6-V_5)
V_{16}= logmonitor_wn_Time	Running+Done	(V_9-V_6)
V_{17}= lrms_wn_Time	Running	(V_8-V_6)
V_{18}= total_Time	Submitted+Waiting+ Ready+ Running+Done	(V_9-V_1)

The detailed definitions of the metrics in Table 1 are:

- **V_{10} = registration time** of a job, defined as the duration between the submission of a job to the LCG/EGEE environment and the time the job is accepted by the network server of the RB node,
- **V_{11} = match making time** of a job, defined as the duration between the acceptance of a job from the network server of the RB node, and the time the WMS service of the RB node finds the appropriate CE for executing the job,
- **V_{12} = ready to transfer to CE time** of a job defined as the duration between the submission of a job to the LCG/EGEE environment and the time the job reaches the appropriate CE and is forwarded to the Gatekeeper of the CE,
- **V_{13} = transfer time** of a job, defined as the duration between the time the job controller of the RB node sends the job for execution to the appropriate CE and the time the job reaches the appropriate CE and is forwarded to the Gatekeeper of the CE,
- **V_{14} = Total CE time** of a job, defined as the duration between the time the CE receives the request and the time the output of the job has been transferred back to the RB node. This time duration corresponds to the time that a job spends at the CE,
- **V_{15} = CE queuing time** of a job, defined as the duration between the reception of request by the CE and the time the user files have been copied from the RB to the WN where the job will be executed,
- **V_{16} = WN execution time (logmonitor)** of a job, defined as the duration between the time the user files have been copied from the RB to the WN where the job will be executed and the time the user can retrieve the output of his job to the UI,
- **V_{17} = WN execution time (lrms)** of a job, defined as the duration between the time the LRMS handles the job execution on the available local farm of worker nodes and the Epoch the output of the job has been transferred back to the RB node,
- **V_{18} = Total time** of a job, defined as the duration between the submission of a job to the LCG/EGEE environment and the time the user can retrieve the output of his job to the UI.

Based on these metrics we define the four main delay components that comprise the job processing in the LCG/EGEE environment; the total time of a job (V_{18}) is the sum of these four delay components.

- **D_1 = V_{12} = ready to transfer to CE time** describes the time the job stays at the Pending, Submitted and Waiting states. This delay component consists of the time a job requires to register with the RB, and the time the RB takes to run the match making service and create the wrapper scripts to transfer the job to the chosen CE.
- **D_2 = V_{13} = transfer time** describes the time the job stays at Ready state. This time consists of the time required to transfer the job wrapper scripts from the RB to the chosen CE.
- **D_3 = V_{15} = CE queuing time** describes the time the job stays at Scheduled state. This time corresponds to the time the job stays at the CE queue before it starts to execute at a WN (including the time that is required to transfer the input user files –input sandbox- from the RB to the that WN).
- **D_4 = V_{16} = WN execution time (logmonitor)** describes the time the job stays at Running and Done states. This time consists of the time required to execute the job and to transfer the output files – output sandbox- to the corresponding RB from which the user can retrieve them. It is worth noting that after the output files have been transferred to the RB the job state becomes and remains Cleared (until the user retrieves the output files or the system discards them). In the definition of the delay components previously presented, we have not considered the time the job stays in the Cleared state since it mainly depends on the user and does not correspond to a quantifiable characteristic.

4. Statistical Results on the LCG Usage

Using the daily reports in ASCII format supplied by the Real Time Monitor tool we acquired information on the traffic submitted to the LCG/EGEE infrastructure and the time durations the jobs spent in each of the processing states before completing execution. The Real Time Monitor (RTM) [3] is a java applet that monitors the LCG in real time. It shows the times at which user jobs are submitted to the Resource Brokers all over the world, the way they are distributed to the sites, and finally, depending on the successful or not execution, the times at which the jobs complete the different states of their processing.

We concatenated the daily ASCII report files and obtained a file that included the desired information in a

form that was suitable for processing using statistical analysis tools. The time period of the observation was one month (starting from 1st of October 2006 until 31st of October 2006). The total number of jobs that were submitted during this period was 2228838.

From the Real Time monitor Tool we were able to retrieve general information regarding the job processing and also the time epochs that correspond to specific events in the LCG/EGEE environment. By manipulating these epochs we were able to calculate the metrics presented in Table 1 and thus analyze the times the job spent at different states of its processing and thus the corresponding delay components.

Table 2 shows the values of the minimum, the maximum, the mean and the standard deviation of the job inter-arrival times, and the metrics (V_{10} to V_{19}) recording the time durations spent by a job at different states in the LCG/EGEE environment.

Table 2: Statistical results for the metrics used. N is the job number from which the results were computed. Min, max, mean and std. deviation are measured in secs

	N	Min	Max	Mean	Std. Dev
Interarrival time	980581	0	60	1.25	1.52
V_{10}= registration_Time	2166574	1	14679	14.9	79.579
V_{11}= match_Time	1824822	1	65794	96.7	841.783
V_{12}= D_1= ready _to_transfer_to_CE_Time	1784806	1	65808	141.4	894.825
V_{13}= D_2= transfer_Time	1767897	1	999822	12411.6	72363.75
V_{14}= D_3+D_4 = logmonitor_CE_total_Time	1365789	2	1099682	39757.3	88809.94
V_{15}= D3= logmonitor_CE_queue_Time	1170688	2	1099673	16899.0	61007.08
V_{16}= D_4= logmonitor_wn_Time	1170804	1	1201163	14454.5	38012.27
V_{17}= lrms_wn_Time	1039674	1	1752808	14248.7	36403.97
V_{18}= D_1+D_2+D_3+D_4 = total_Time	1355887	17	1099957	49286.7	113684.6

Job inter-arrival times

Figure 3 illustrates the cumulative distribution function (cdf) of the inter-arrival times of the jobs submitted to the LCG/EGEE infrastructure. It is worth noting that the Real Time Monitor tool, from which we obtained the measurements, stores the corresponding time instances in seconds, which means that the real time values are rounded to the closest integer second. This determines the accuracy of our observations. We can observe that with high probability (around 0.4) the inter-arrival time between two jobs is close to 0 sec (the inter-arrival times represented as 0 sec include the inter-arrival times up to 0.5 sec). The maximum observed value was 60 sec, and the probability of observing an inter-arrival time greater than 7 sec is negligible. Since the inter-arrival times' standard deviation is quite small and close to its mean (Table 2) we can assume that the inter-arrival process is quite close to a Poisson process.

Registration, Match-making, Ready to transfer to CE and Transfer times

In this section, we present results regarding the: Registration (V_{10}), Match-making (V_{11}), Getting ready to transfer to CE ($V_{12} = D_1$), and Transfer ($V_{13} = D_2$) times.

From Figure 4 we observe that the match making times and getting ready to transfer to CE times exhibit similar behaviors, and the majority of the observed values lies in the range of a few seconds to a few tens of seconds, as can be deduced from the steep step-like form of the cumulative distribution function (cdf) in that region. Registration time has a small probability (~ 0.06) to be less than 5 sec and a high probability (~0.9) to be between 6 and 50 sec. Match making time has a small probability (~0.07) to be less than 7 sec and a high probability (~0.85) to be between 8 and 66 sec. Getting Ready to transfer to CE time includes the registration time (Pending state), the match making time (Submitted state) and an additional delay in which the RB creates a wrapper script and prepares the job for submission to the chosen CE (Waiting state). Since the match making time dominates the two other delay components, getting ready to transfer to CE times cdf is similar to the cdf of the match making times shifted by a few seconds (10 to 100). This observation can also be verified by comparing the mean and standard deviation of the getting ready to transfer to CE times with those of the match making times -Table 2 (their mean values differ by 50 sec while the values of their standard deviation are almost equal).

From Figure 4 we see that the probability of observing a transfer time smaller than 3 sec is small (~0,06), while the probability of observing a value less than 80 sec is high (~0,84). However, from the transfer times cdf we can see that this variable seems to exhibit a heavy tail and there is also a considerable probability (~0.16) of observing values in the range of hundreds to millions of sec. The difference of the transfer times (heavy tail) with the variables analyzed in the previous paragraph can be also verified by the large value of the transfer times' standard deviation (Table 2).

CE Queuing, WN Execution and Total CE times

In this section, we present results regarding the delay introduced at the Computing Element (CE) of an LCG/EGEE cluster. More specifically, we present results for the CE Queuing ($V_{15} = D_3$), the logmonitor WN Execution ($V_{16} = D_4$), the lrms WN Execution (V_{17}), and the CE Total ($V_{14} = V_{15}+V_{16} = D_3+D_4$) times.

Comparing Figure 5 and Figure 4 we observe that the cdf of the variables presented in this section increase less rapidly than the cdf of the variables presented in the previous paragraph. The results of Figure 5 indicate that a job queuing time starts from 100 sec and have a high probability to be less than 200 sec. However, queuing times can also take large values and even reach 10^6 sec.

The logmonitor WN times and the lrms WN times differ only slightly for values less than 1000 sec (specifically lrms WN times have a higher probability to

take smaller values) and converge for large values. They appear with equal probability (~0,56) to be less than 1000 seconds, and can reach values of 10^6 sec. Note that the difference between these two variables (logmonitor WN – lrms WN) corresponds to the time a job spends in the Done state, which is the time required to transfer the output sandbox from the CE to the RB, indicating that the output sandbox requires only a small amount of time to be transferred. CE total time includes the queuing and logmonitor WN time. There is a medium probability (~0,35) to observe a CE total time less than 1000 seconds, while this variable can reach values of the order of 106 sec. The mean value of the CE total times was measured to be equal to $38.75 \cdot 10^3$ sec and its standard deviation was $88.8 \cdot 10^3$ sec.

Total times

The results in Figure 6 indicate that the total times ($V_{18} = D_1+D_2+D_3+D_4$) of the jobs exhibit almost similar behavior with the CE total times (CE queuing + WN execution times = D_3+D_4). CE total times dominate the total times, while getting ready to transfer to CE times (D_1) and transfer times (D_2) contribute negligibly to overall delay. The job total times are between 200 and 105 sec with probability ~0.91, and can also take large values (10^7 sec).

5. Modeling of the Interarrival Times and the Delay Components of a Job

In this section we are interested in modeling the job arrival process and the delay components incurred by a job in the LCG/EGEE environment. As delay components we consider the four delay components introduced in Section 3 and analyzed in Section 4.

5.1. Modeling the job arrival process

Based on the descriptive statistics (Table 2) and the cumulative distribution function of the inter-arrival times (Figure 3) we want to characterize the overall job arrival process in LCG/EGEE. Since the standard deviation of the inter-arrival times is quite close to its mean and, from the corresponding cdf, they do not seem to exhibit a heavy tail, a Poisson process is quite likely to model the arrival process behavior. We have experimented with exponential distributions and parameters close to $^1/_{observed\_mean}$. Figure 7 shows the cdf of the inter-arrival times and the cdf of an exponential distribution with mean 1.6077 sec. It is worth noting that the observed values were integers (our observations were rounded to the closest second). Therefore, in order to fairly compare the two distributions we have rounded to the closest integers the values produced by the proposed exponential distribution (referred to as rounded exponential model). After this adjustment, the exponential distribution with mean 1.6077 sec resulted in a distribution with mean 1.15 sec and standard deviation 1.57. We can observe that the

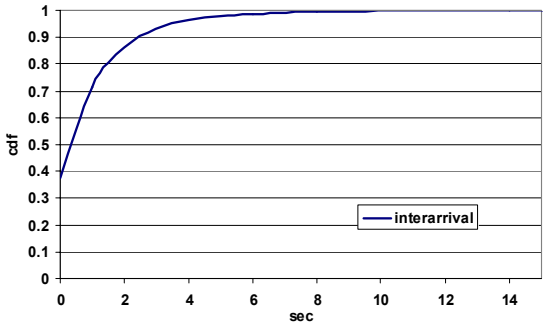

Figure 3 - Empirical cdf of the interarrival times.

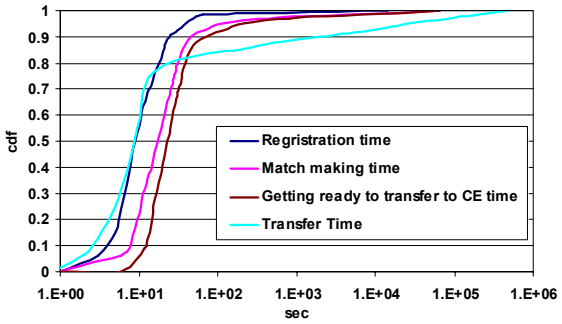

Figure 4 - Empirical cdf's of the Registration times (V_{10}), Match making times (V_{11}), Ready to transfer to CE times ($V_{12} = D_1$) and Transfer times ($V_{13}=D_2$).

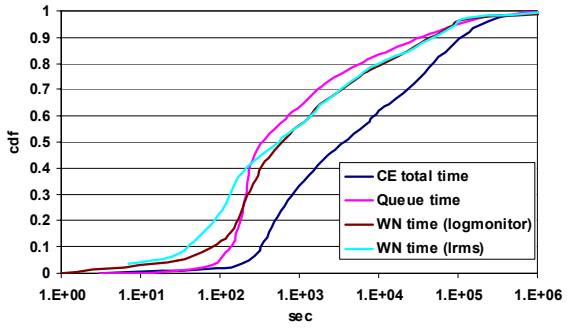

Figure 5 - Empirical cdf's of the Total CE times (V_{14}), and the components that comprise it. We plot the cdf's of the CE Queuing times ($V_{15} = D_3$) and the WN Execution times according to logmonitor ($V_{16} = D_4$) and lrms (V_{17})

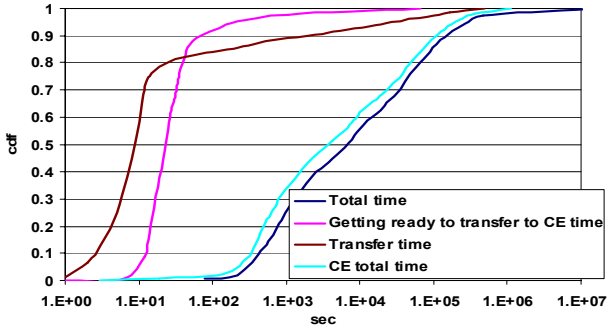

Figure 6 - Empirical cdf's of the Total job times (V_{18}), and the constituent delays that comprise it. We plot the cdf's of the Ready to transfer to CE times (V_{12}), the Transfer times (V_{13}) and the CE Total times (V_{14})

rounded exponential distribution can adequately model the job arrival process in the LCG/EGEE environment.

5.2. Getting Ready to Transfer to CE times (D_1) modeling

From Table 2 we observe that delay component D_1 exhibits the smallest standard deviation among the delay components defined in Section 3. D_1 corresponds to the time a job stays at the Pending, Submitted and Waiting states and thus is the time that the jobs spends in the UI and the RB before being transferred to a cluster. From Figure 4 we observe that D_1 takes with high probability values close to its mean 141.44 sec. Moreover, from Figure 6 we see that the job total delay is dominated by CE times (D_3 and D_4). Therefore, we regard that the modeling of the getting ready to transfer to CE delay component (D_1) as a constant (equal to its mean 141.44 sec) is an acceptable approximation, since in any case it contributes the smallest delay to the total delay.

5.3. Transfer times (D_2) modeling

The Transfer times ($V_{13} = D_2$) presented in Figure 4, as well as the CE queuing times ($V_{15} = D_3$) presented in Figure 5, exhibit linear behavior at different stages (with different slopes) in the logarithmic scale. We investigate how a hyper-exponential process (in the general category of phase type distributions) and a phase lognormal distribution can fit the behavior of these delay components. We examined these two alternatives since the hyper-exponential distribution is widely used for modeling, while the phase lognormal distribution seems appropriate to model the linear behavior observed at different stages in the logarithmic scale.

The probability density function (pdf) of an m-phase hyper exponential random variable (r.v.) X is given by:

$$f_X(x) = \sum_{i=1}^{m} p_i \cdot f_{Ei}(e) = p_1 \cdot f_{E1}(e) + p_2 \cdot f_{E2}(e) + ... + p_m \cdot f_{Em}(e)$$

where E_i is an exponential r.v. with mean $1/\lambda_i$, and p_i is the probability that X takes on the form of E_i (thus, $\sum_{i=1}^{m} p_i = 1$). Similarly, the pdf of an m-phase lognormal r.v. X is given by the same equation, but this time E_i is a lognormal r.v. with average a_i and standard deviation d_i (a r.v. L_i follows the lognormal distribution if the r.v. $ln(L_i)$ is normally distributed), and p_i is the probability that X will take on the form of L_i.

Regarding the modeling of the Transfer times ($V_{13} = D_2$), we considered three alternatives: (i) a 3-phase hyper-exponential (H_3), (ii) the sum of a deterministic and a lognormal r.v. and (iii) a 2-phase lognormal distribution.

We chose to use two phases for the lognormal model driven by the observation that Figure 4, exhibits linear behavior in two different periods in the logarithmic scale. For the hyper-exponential model we used three phases driven by the observation that Figure 4 exhibits one noticeable step and also has a heavy tail (assuming that we need one phase to model the step and at least two

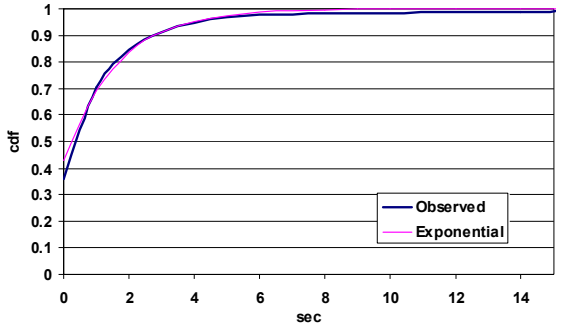

Figure 7 - Cdfs of the interarrival times of the actual observations and the examined exponential model

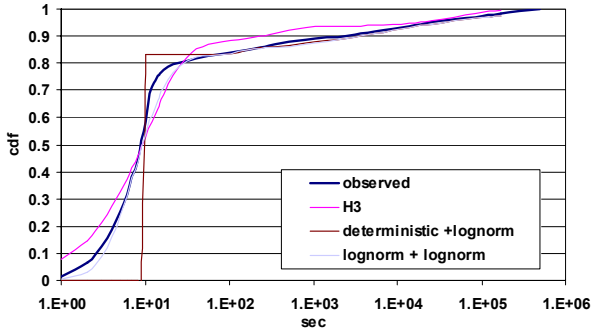

Figure 8 - Cdfs of the transfer times of the actual observations and the examined models

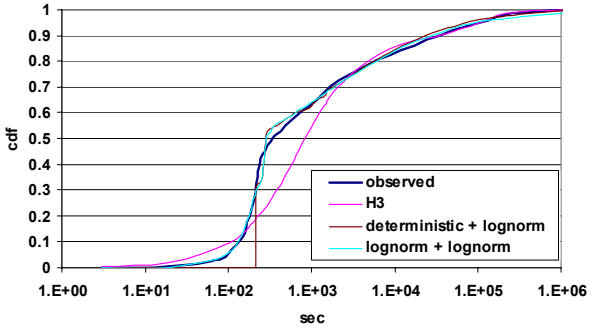

Figure 9 – Cdfs of the CE queuing times of the actual observations and the examined models

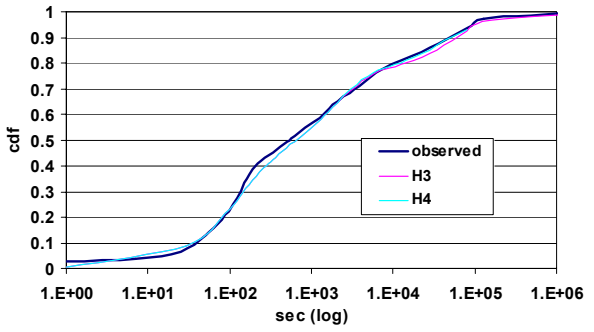

Figure 10 - Cdfs of the WN execution times of the original observations and the examined 3- and 4- phase Hyper-exponential model.

phases to model the heavy tail). For the hyper-exponential model we used the EMpht utility [12] to obtain the corresponding parameters.

The parameters that provide the best fits of D_2 with the three models examined were found:

- Case (i) p_1=0.8635, p_2=0.0711, λ_1= $9.377 \cdot 10^{-2}$ sec^{-1}, λ_2= $2.959 \cdot 10^{-3} sec^{-1}$, and λ_3= $1.4 \cdot 10^{-5}$ sec^{-1},
- Case (ii) p_1=0.83, constant=9, lognormal average=8.8126 sec, std deviation = 3.1227 sec,
- Case (iii) p_1=0.83, a_1= 2.027 sec, d_1=0.7380 sec, a_2=8.8126 sec, d_2 = 3.1227 sec.

Figure 8 shows the empirical cdf of the job Transfer time**,** as presented in Section 4, and the cdfs we obtained for the proposed models. We can observe that the 2-phase lognormal distribution is the more accurate model while the hyper-exponential and the sum of a deterministic and a lognormal distribution converge to the observed data only for large values (heavy tail). Since, in general, the heavy tail dominates the performance of this delay component, these two alterna-tives can be also considered acceptable approximations.

5.4. CE queuing times (D_3) modeling

We considered again three alternatives for modeling the CE queuing times (V_{15} = D_3): (i) a 3-phase hyper-exponential model (H_3), (ii) the sum of a deterministic and a lognormal r.v. and (iii) a 2-phase lognormal distribution. The corresponding parameters were found:

- Case (i) p_1=0.619, p_2=0.2408, λ_1= $1.536 \cdot 10^{-3}$ sec^{-1}, λ_2= $2.71 \cdot 10^{-4}$ sec^{-1}, and λ_3= $1.2 \cdot 10^{-5}$ sec^{-1},
- Case (ii) p_1=0.32, constant=210 sec, lognormal average=7.1093 sec, standard deviation = 2.85 sec,
- Case (iii) p_1=0.34, a_1= 5.13 sec, d_1=0.211 sec, a_2=7.1093 sec, d_2 = 2.85 sec

Figure 9 shows the empirical cdf of the job CE queuing time as presented in Section 4 and the cdfs we obtained by the proposed models. Similar to D_2, the 2-phase lognormal distribution seems to be the best model, while the other two models are also good approximations.

5.5. WN Execution times (D_4) modeling

The WN execution times (V_{16} = D_4), as presented in Section 4 (Figure 5), exhibit peaks at certain periods. We investigated how a hyper exponential random variable can fit this behavior. We used only this type of process since it is widely used in the literature to model execution times. More specifically, we considered two cases: (i) a 3-phase (H_3), and (ii) a 4-phase (H_4) hyper exponential distribution. We chose to use these values for the number of phases driven by the observation that Figure 5 exhibits 3-4 steps. We used again the EMpht utility [12] to obtain the corresponding parameters:

- Case (i) p_1=0.3888, p_2=0.3635, λ_1= $8.031 \cdot 10^{-3}$ sec^{-1}, λ_2= $5.47 \cdot 10^{-4}$ sec^{-1}, and λ_3= $1.46 \cdot 10^{-5}$ sec^{-1}, and
- Case (ii) p_1=0.3776, p_2= 0.3614, p_3=0.1199, λ_1= $9.021 \cdot 10^{-3}$ sec^{-1}, λ_2= $5.52 \cdot 10^{-4}$ sec^{-1}, λ_3= $1.359 \cdot 10^{-5}$ sec^{-1}, λ_4= $1.559 \cdot 10^{-5}$ sec^{-1}.

Figure 10 shows the empirical cdf of the job WN execution time as presented in Section 4 and the cdfs obtained for the two models. Since the accuracies obtained by the 3- and 4-phase models are similar, we conclude that a 3 phase hyper exponential process is sufficient for modeling the WN execution times.

6. Conclusions

A thorough analysis of the job arrival process and the time durations jobs spend at different states in the LCG environment was presented. The job inter-arrival times were found to match very well with a rounded exponential distribution. We defined four delay components of the total job delay, and proposed and validated probabilistic models for each component separately. We observed that the total time a job stays in the LCG environment is dominated by the Computing Element's Queuing delay and the Worker Node's execution time.

7. Acknowledgment

This work has been supported by the EC through the Phosphorus and the e-Photon/One+ IST projects. The authors would like to thanks Gidon Moont from Imperial College London for his help with the logs of the Real Time Monitor.

8. References

[1] I. Foster, C. Kesselman, "The Grid: Blueprint for a New Computing Infrastructure", 2nd Edition, Morgan Kaufman

[2] The EGEE project homepage: http://public.eu-egee.org/

[3] Real Time Monitor: http://gridportal.hep.ph.ic.ac.uk/rtm/

[4] E. Medernach, "Workload analysis of a cluster in a Grid environment", Proc. of 11th JSSPP workshop, 2005.

[5] H. Li, M. Muskulus and L. Wolters, "Modeling Job Arrivals in a Data-Intensive Grid", Proc. of 12th JSSPP workshop, 2006.

[6] gLite-3 user's guide, https://edms.cern.ch/file/722398//gLite-3-UserGuide.pdf

[7] http://goc.grid.sinica.edu.tw/gstat/index.html

[8] http://www.eu-egee.org/

[9] W. Cirne and F. Berman, "A comprehensive model of the supercomputer workload", Proc. of 4th IEEE annual workshop on workload characterization, 2001.

[10] B. Song , C. Ernemann, R. Yahyapour, "Parallel Computer Workload Modeling with Markov Chains", Proc. 10th JSSPP workshop, 2004.

[11] https://lcg-registrar.cern.ch/virtual_organization.html

[12] EMpht: http://home.imf.au.dk/asmus/pspapers.html