Characterizing E-Science File Access Behavior via Latent Dirichlet Allocation

Yusik Kim, Cecile Germain-Renaud

Characterizing E-Science File Access Behavior via Latent Dirichlet Allocation

Yusik Kim
INRIA-Saclay
LRI Bât. 490, Université Paris-Sud 11
Orsay, France
Email: kim@lri.fr

Cécile Germain-Renaud
Université Paris-Sud 11
LRI Bât. 490, Université Paris-Sud 11
Orsay, France
Email: cecile.germain@lri.fr

***Abstract*—E-science is moving from grids to clouds. Getting the best of both worlds needs to build on the experience gained by the steady operation of production grids since some years. With the Grid Observatory initiative, trace data are publicly available to the computer science and engineering community and can be used for dimensioning and optimizing infrastructure.**

This paper proposes a new approach for analyzing behavioral traces: as most of them are indeed text documents, state of the art techniques in text mining, and specifically Latent Dirichlet Allocation, can be exploited. The advantages are twofold: providing some level of explanation inferred from the data; and a relatively scalable way to capture the temporal variability of the behavior of interest, while retaining the full dimensionality of the problem at hand.

We experiment the text mining analogy approach by characterizing file access behavior. We validate the resulting probabilistic model by showing that it is capable of generating synthetic traces statistically consistent with the real ones.

***Index Terms*—Graphical Models; Trace Analysis; e-science infrastructures**

I. Introduction

In the last ten years, scientific communities worldwide have pioneered the deployment of extreme e-science computational infrastructures, in the form of grids. At the European level, the High Energy Physics (HEP) community has been the driving force for building the largest existing non-profit system, the EGEE/EGI grid [1]. This real-world, production-quality grid experience is relevant for cloud research and engineering for three reasons. Firstly, computational grids provide new natural examples of emerging collective behavior, the distinctive feature of complex systems; second, at least for the EGEE/EGI grid, monitoring data are publicly available for research through the Grid Observatory initiative [2]; finally, e-science is moving to cloud technology, with its own requirements.

This paper focuses on the usage pattern as characterized by file access. Analyzing this behavior as well as the ability to explain it through mathematical models would be interesting in its own right but it would also prove to be an important asset for system administrators who wish to use this knowledge to monitor and/or regulate the usage of the resources. The implications of characterizing, for example, the access pattern of files shared by e-science users include realistic requests modeling, inferring latent relationships among files and users, anomaly detection, reducing file I/O latency by optimizing caching policies, etc.

The goal of this paper is to create and validate a generative model of these accesses. By definition, a generative model can generate synthetic realizations simulating realistic usage. In this work, the model is derived from the transaction traces provided by the GridFTP protocol. In a first attempt, we tried to discover descriptive characterizations of classical complex systems indicators, such as popularity, locality, and social graphs parameters. Overall, the resulting statistical distributions do not follow simple models such as a power law. Going further needs to disentangle a network of causal interactions between the users and the middleware, which cannot be defined a priori.

To propose a probabilistic model capable of explaining how the observed responses could have been generated, our key observation is that a trace can be considered a *text document*. Characterizing a text document based on the observed words is central to text mining, and has been extensively studied [3]. From this perspective, a document is described in terms of unobservable (*latent*) *topics*, which are in turn described in terms of observable *words*: the topics relevant to the document generate all the words in the document. Latent Dirichlet Allocation (LDA) [4] is a well established probabilistic model for this class of causal structure.

As far as we know, no previous work has considered LDA for behavioral modeling of large distributed systems. The advantages of this approach are twofold. The first one is that the topics - in our context, the unknown causes - are inferred from the data, not defined a priori. Second, and more specifically to trace analysis, LDA offers a relatively scalable way to capture the temporal variability of the behaviors of interest (here the file accesses), while retaining the full dimensionality of the problem at hand, and making limited and well-defined a priori assumptions, essentially that the user request probabilities over successive periods are sampled i.i.d. from a Dirichlet distribution with some unknown parameter. The main result of this paper is that LDA is indeed instrumental for trace modeling.

The rest of this paper is organized as follows. Section II provides an empirical analysis of a 11 week trace from a major EGI site. Section III fits two generative models first

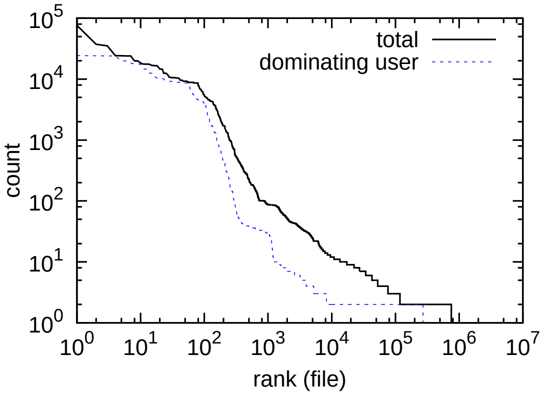


Fig. 1: File popularity.

for a generic LDA model, and second by taking advantage of the supplementary information on causes that are present in the trace. Some of the statistical properties revealed in the empirical study are exploited to make this analysis tractable. This section also develops the above-mentioned advantages compared with a classical time series approach. Section IV experimentally validates the model. Section V briefly discusses related work, before the conclusion.

II. Empirical Study on File Access Patterns

The data set, publicly available from the Grid Observatory portal, spans a 11 week period from 29/11/2010 to 31/02/2011 including 5,204,269 transactions by 262 distinct users on 2,123,734 distinct files. The monitored storage is the one of the LAL site; however, the requests can originate from across the whole EGI grid, because its storage is fully distributed. The next question is to which extent this trace is representative of the EGI file traffic. This is a difficult issue, on which we will get back in Section V and in the conclusion. A specificity of the trace is that is it truly multi-disciplinary, with significant usage from the Complex Systems community, the HEP experiments, and Cosmology.

The primary goal was to characterize the measurements of interest in terms of known parametric distributions. The metrics (first column of table I) will be described later; for now, we see that in all cases except one, the standard deviation is larger than the mean by 1 or 2 orders of magnitude, requiring to study the complete distributions.

A. File Popularity and User Behavior

With the prevalence of the power law distribution observed in popularity based measures (e.g., city populations within a country, outdegree of sites on the Internet, frequency of words within a text, etc.), we expected a similar behavior for count-based measures such as file popularity, user activity, and outdegree of user-file relation graphs. However, it turns out that this is not the case.

We measure the popularity of a file that resides in the grid during the study period using three different metrics: file access count, user and file outdegree.

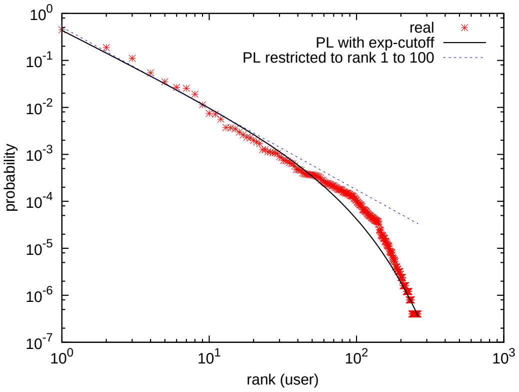


Fig. 2: Distribution of user outdegree. MLE $\alpha = 1.577, \gamma = 0,020$ for the PL with exponential cutoff, and $\alpha = 1.732$ for the PL restricted to the upper tail.

The first metric, the *file access count*, is the total number of requests for a file, including multiple requests by the same user. This metric measures the popularity from a workload perspective and is sensitive to usage artifacts: for example, if a job repeatedly reads the same file without copying it first to local space (which would be the sensible strategy), that particular file may show a surge in popularity. The rank-frequency plot on a doubly log scale (Figure 1) shows that the popularity distribution does not seem to resemble any known parametric distribution including the power law (here we do not normalize the count in order to show the magnitudes). A more detailed analysis shows that the activity is dominated by only a few users where less than 2% of the users account for nearly 80% of the requests (for the sake of completeness, it must be said that in fact, at least some of these dominating users are generic ones, acting as a proxy for real users, in a portalized usage of the grid). The rank-frequency plot restricted to the most active user, also on Figure 1, shows that the shape of the popularity distribution is highly influenced by the behavior of the dominating user. We believe that this dominance in the overall access count by a few users is what prevents the file popularity from exhibiting a power law distribution.

The basic representation of the requests structure is a bipartite graph with the distinct users as the nodes on one side, the distinct files as the nodes on the other side; there is an edge between a user and a file if the user requested the file at least once.

The second metric is the *file outdegree* (the number of distinct users who accessed it). This metric measures the popularity of a file from a social perspective, i.e., how diverse its userbase is. Due to space limitation, we only give the conclusion: the files are well segregated with respect to the users, i.e., each file has a small number of users, at most 7, and most have only one user.

The third metric is the *user outdegree* (how many different files the user accessed), and characterizes the behavior of a user. Figure 2 plots the outdegree distribution of the user

	Sample size	Mean	Std	Median	Min	Max
File access count	2123734	2.45	103.61	1	1	76025
User outdegree	262	9537	77561	105	1	1123645
File outdegree	1002186	1.17	0.39	1	1	7
User access count	262	19863	163013	128	1	2486709
Lifetime (sec)	754887	339699	937056	11259	0	6651860
Cluster size	63	4.16	21.15	1	1	170

TABLE I: Summary statistics

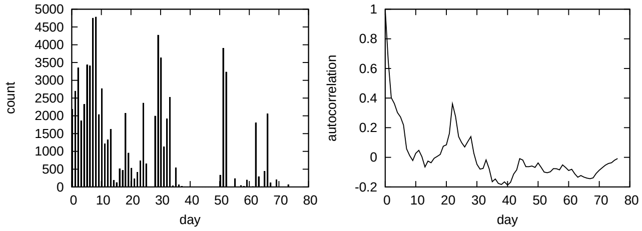


Fig. 3: Daily access count for the most popular file. Left graph: histogram. Right graph: autocorrelation.

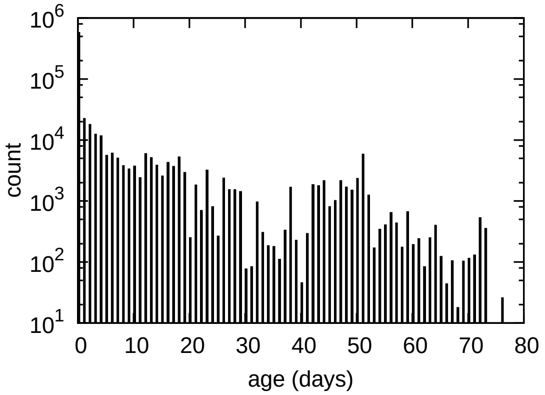


Fig. 4: File lifetime distribution.

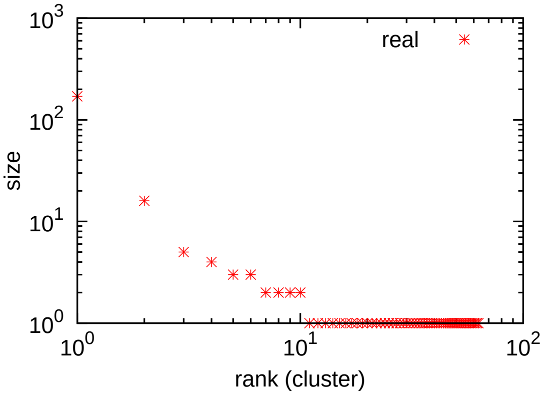


Fig. 5: Cluster size distribution.

nodes. On initial observation, a power law distribution with exponential cutoff, $p(x) \propto x^{-\alpha}e^{-\gamma x}$, is plausible. The maximum likelihood estimation of the parameters was computed with the code of [5], and the goodness of fit evaluated with a χ^2 test. The resulting p-value is 2.2×10^{-16}, thus invalidating the fit. We also tried restricting the fit to the upper tail part (i.e., from rank 1 to rank x_{max}), hoping to reveal at least a local power law behavior, to no avail, as for $x_{max} = 3, ..., 262$ the p-value computed from the χ^2-test was 0.

B. Temporal locality of file and user access

Figure 3, left graph, depicts the number of daily accesses to the most popular file. We observe a recurrent pattern. To better measure the temporal locality of accesses, we examined the empirical autocorrelation of the daily access frequency vector. Figure 3, right graph, shows a drop to near zero at a 7 day lag, which will have implication for the LDA model.

The recurrent access of the most popular file however, is not representative of the other, less popular, files. Figure 4 shows the distribution of the files' lifetime, defined as the difference between its first and last access date. Many files seem to be more transient, with a shorter lifetime. While this conclusion might not be absolute, a 11 week window is a reasonable time length for practical planning and analysis purposes.

C. Clustering the Users

To get a more concise representation of the trace, it makes sense to cluster the users with similar interests in terms of file access. File grouping, on the other hand, has been proposed in several studies for data staging [6] and for data visualization [7]. Ideally, we would like to group the users along a common property. Co-access is the simplest one. The interpretation of the resulting clusters is outside the scope of this work, as it depends heavily on external factors such as the type of applications: for example, if a research project is of the type single instruction multiple data, two collaborators may never access the same file, in which case having co-access to a file may indicate a competitor relationship.

The strength of the link between any two users is the number of common files that both accessed. Connecting two users who have link strength of at least 1, the resulting graph will be partitioned into connected components, which are the requested clusters. In this way, the 262 distinct users are partitioned into 63 clusters. The largest cluster contains 170 users and there are 10 clusters with more than 1 user (Figure 5). We later use the number of clusters as a parameter of the LDA model.

III. Generative Models for Access Pattern Characterization

A. Overview

The previous empirical study describes what the current state of the system looks like during the analyzed 11 week period, but provides little insight on why we see what we see because it is merely a response to whatever (possibly latent) cause that generates the transactions. The goal of this section is to propose a probabilistic model that is capable of explaining how the observed responses could have been generated.

The number of transactions made by each user during the 11 week period indicates the contribution each user had on the overall behavior of the system. However, user A accounting for 50% and user B 30% of the total transactions over the 11 week period does not imply that they had the same ratio for, say, week 2. In fact, the individual user contributions fluctuate within the 11 week period to arrive at cumulative figures like 50% and 30% for the whole period. There are a few options for modeling this time variability. A naive approach would be to ignore the time variability and use the cumulative figures for each period. If we used this approach for building a model to generate synthetic trace, each period will look nearly identical, which is clearly not desirable. A second approach would be to consider the proportion of transactions made by each user within each individual week as a random vector and deal with the sequence as a stochastic process. We need to decide on how to model this multi-dimensional stochastic process. In this paper, we assume that this stochastic process is i.i.d.

Our specific model choice is influenced by the observation that there is an analogy between the transaction traces and text documents. In text document characterization, an important goal is to describe a document in terms of unobservable topics through observable words. A document is then a mixture of topics, and characterizing the document amounts to identifying the topic mixture probability.

Putting our problem in the same framework, our goal is to characterize the transaction trace based on the observed file accesses. Thus, we make the assumption that there are several "causes" which themselves are characterized by a specific file access pattern. We can use this analogy to take advantage of the machinery of LDA developed for text documents. Once we model our problem as a LDA, the remaining technical problem is to estimate the model parameters.

B. Latent Dirichlet Allocation

To be reasonably self-contained, this section formally sketches LDA [4]. The most intuitive description is in the text mining context where we wish to explain how words are generated within a document. LDA assumes a conditional distribution of words given a topic and that a document is characterized by its weights on different topics. It is further assumed that the topic weights for each document follow a common Dirichlet distribution. The generative process works as follows: words within a document are generated by first sampling a topic according to the weights and then sampling a word conditioned on the sampled topic.

Having this basic structure in place, it remains to specify the conditional distributions that are involved, which requires. to introduce quite a lot of notations. Let K be the number of topics, M the number of documents, and L the size of the vocabulary. We denote:

α The K dimensional Dirichlet parameter. The i-th component is denoted by α_i.

θ The K dimensional vector of probabilities sampled from Dirichlet(α) representing the topic distribution within a document. The i-th component is denoted by θ_i.

For each word in a N words document, we denote:

u A scalar representing the chosen topic in $\{1, ..., K\}$

u^i The indicator function $u^i = \mathbf{1}\{u = i\}$.

f A scalar representing the chosen word in $\{1, ..., L\}$

f^i The indicator function $f^i = \mathbf{1}\{f = i\}$.

β The $K \times L$ matrix whose rows represent the word selection probabilities conditioned on each of the K topics. $\beta_{ij} = p(f^j = 1 | u^i = 1)$.

Here we have intentionally suppressed the dependence on the document index m for θ and N, and both the word index n and document index m for u and f to simplify notation. When necessary, we will use the full notation N_m, θ^m, u_{nm}, and f_{nm} to make the dependence explicit.

LDA can be described as the following generative process for each of the M documents.

1) Choose $N \sim$ Poisson(ξ).
2) Choose $\theta \sim p(\theta|\alpha)$, a Dirichlet distribution.
3) For each of the N words within the document:
   a) Choose a topic $u \sim p(u|\theta)$, a multinomial distribution.
   b) Choose a word $f \sim p(f|u, \beta)$, a multinomial distribution.

Finally, each conditional distribution is defined as follows:

- $p(\theta|\alpha) = B(\alpha) \prod_{i=1}^{K} \theta_i^{\alpha_i - 1}$, where $B(\alpha)$ is the coefficient of the pdf of Dirichlet(α).
- $p(u|\theta) = \prod_{i=1}^{K} \theta_i^{u^i}$
- $p(f|u, \beta) = \prod_{i=1}^{K} \prod_{j=1}^{L} \beta_{ij}^{u^i f^j}$

The graphical model representation of LDA is illustrated in Figure 6. For the readers not familiar with graphical models, these figures state that the joint distribution of a single document given the parameters α and β is

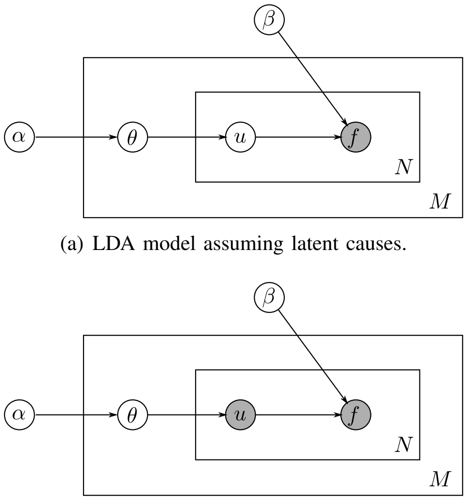
$$p(\theta, \mathbf{u}, \mathbf{f} | \alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(u_n|\theta) p(f_n|u_n, \beta) \qquad (1)$$

where $\mathbf{u} = (u_1, ..., u_N)$ and $\mathbf{f} = (f_1, ..., f_N)$. And the joint distribution for the corpus of M documents is

$$p(\Theta, \mathbf{U}, \mathbf{F} | \alpha, \beta) = \prod_{m=1}^{M} p(\theta^m, \mathbf{u}_m, \mathbf{f}_m | \alpha, \beta). \qquad (2)$$

C. LDA applied to transaction traces

We apply LDA to the problem of modeling file access behavior on a grid computing platform by using the following analogy between a text corpus and a transaction trace.



(a) LDA model assuming latent causes.

(b) LDA model assuming causes are observed.

Fig. 6: Graphical model representation of LDA. Observed data is represented by shading the node, which we later condition on to find the posterior distribution of the latent variables.

- Word translates to Filename
- Topic translates to Cause
- Document translates to a Trace snippet
- Corpus translates to the Complete trace

The term "cause" may need further clarification. It can be thought of a category which is characterizable by its file access pattern. For example, causes can be categories of job types (simulation job, equation solver, etc.), fields of study (particle physics, plasma physics, astrophysics, etc.), research projects, or Virtual Organizations, which is the fundamental administrative structure of e-science production grids .

A trace snippet is simply the segment of the trace corresponding to a time window. Using consecutive windows, the collection of all snippets covers the whole trace. Segmenting the trace into snippets is the fundamental trick that makes modeling the temporal variability tractable.

These analogies implicitly assume that 1) each cause is characterized by a file access distribution, 2) each window is characterized by a cause mixture distribution, and 3) the snippets, each of which is characterized by a distribution over files, are mutually independent. The first assumption is inherent to our approach, which exploits only the transaction trace. The second assumption is clearly a rough approximation: ideally, the segmentation should be inferred from the data too, as for instance in [8], [9]. The last assumption is partially justified by the observation of Section II-B, showing that correlation vanishes quite fast, in which case, the only correlation will occur near the boundaries of the periods.

Another important assumption made in the LDA model is that the distribution of words (files) within a document (trace snippet) is exchangeable, i.e., $p(f_1, ..., f_N) = p(f_{\pi(1)}, ..., f_{\pi(N)})$ for any permutation π. This means that the generative process says nothing about the exact order in which files are accessed. This assumption poses no serious problem since we are mostly interested in the aggregate statistics, such as file access frequency or distinct file accesses, within a single window.

With the joint distribution Eq.(2) available, the next step is to estimate the parameters of the model given the observed data, namely the file accessed in each transaction. Recall that the individual causes (u) as well as its mixture probabilities (θ) are latent in our model.

1) Parameter estimation: The procedure for finding the maximum likelihood estimator (MLE) of α and β is covered in [4].

In order to find a maximum likelihood estimate of the parameters, we need an expression for the likelihood function. Integrating over the latent variables θ and $\mathbf{u}$ of Eq.(1):

$$\begin{aligned} p(\mathbf{f}|\alpha, \beta) &= \int p(\theta|\alpha) \prod_{n=1}^{N} \sum_{u_n} p(u_n|\theta) p(f_n|u_n, \beta) d\theta \\ &= B(\alpha) \int_{\sum_i \theta_i = 1} \left(\prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \right) \prod_{n=1}^{N} \sum_{i=1}^{K} \prod_{j=1}^{L} \theta_i \beta_{ij}^{f_n^j} d\theta, \end{aligned}$$

which is intractable to evaluate [10].

A difficulty when trying to use an EM algorithm to compute the MLE of the parameters of this model is that the parameters are coupled which makes it intractable to compute the posterior distribution of the latent variables. Therefore, a *variational EM* is used to approximate the joint distribution so that the parameters decouple, and consequently an approximated posterior distribution of the latent variables can be computed. At each step, the parameters of the approximate distribution is chosen so that the Kullback-Liebler divergence to the true distribution is minimized. Through the variational EM, we get the MLE of the model parameters α and β.

2) When causes are observable: We have so far assumed that causes are latent. However, the transaction trace provides a precious information: the user who requested the file transaction. Therefore, we may explore a simplifying assumption, namely that the individual users of the grid are in fact the causes, effectively making the causes observable. The only remaining latent variable in this case is the cause mixture distribution θ. Let us however pretend for now that θ is observed.

For each period, pretending that θ is observed, and since θ and f are conditionally independent given u, we have the complete likelihood

$$p(\mathbf{u}, \mathbf{f}, \theta|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(u_n|\theta) p(f_n|u_n, \beta).$$

But since the random vector θ is latent, we must integrate over it to get the true likelihood function:

$$\begin{aligned} p(\mathbf{u},\mathbf{f}|\alpha,\beta) &= \int p(\mathbf{u},\mathbf{f},\theta|\alpha,\beta)d\theta \\ l(\alpha,\beta) &= \log \int p(\mathbf{u},\mathbf{f},\theta|\alpha,\beta)d\theta \\ &= \log \int q(\theta|\mathbf{u},\alpha)\frac{p(\mathbf{u},\mathbf{f},\theta|\alpha,\beta)}{q(\theta|\mathbf{u},\alpha)}d\theta \\ &\geq \int q(\theta|\mathbf{u},\alpha)\log p(\mathbf{u},\mathbf{f},\theta|\alpha,\beta)d\theta \\ &\quad - \int q(\theta|\mathbf{u},\alpha)\log q(\theta|\mathbf{u},\alpha)d\theta. \end{aligned} \tag{3}$$

The inequality is a result of Jensen's inequality and q can be any distribution of our choice. We seek to maximize the likelihood function indirectly through maximizing a lower bound to it. When q is fixed, maximizing the provided lower bound of the log-likelihood with respect to α, β reduces to maximizing

$$\mathbf{E}_q[\log p(\mathbf{u},\mathbf{f},\theta)|\alpha,\beta] = \int q(\theta|\mathbf{u},\alpha)\log p(\mathbf{u},\mathbf{f},\theta|\alpha,\beta)d\theta.$$

It can be easily verified that the q that closes the inequality gap is in fact the posterior probability $p(\theta|\mathbf{u},\mathbf{f},\alpha,\beta) \propto \prod_{i=1}^{K}\theta_i^{\sum_{n=1}^{N}u_n^i+\alpha_i-1}$. So we set

$$q(\theta|\mathbf{u},\alpha) = B(\alpha')\prod_{i=1}^{K}\theta_i^{\alpha_i'-1}$$

where $\alpha_i' = \sum_{n=1}^{N}u_n^i + \alpha_i$.

The EM algorithm for finding the MLE of α is proposed in [11]:

- E Step (For each period m):

$$q_m^{(t+1)}(\theta) = B\left(\alpha'^{(t)}\right)\prod_{i=1}^{K}\theta_i^{\sum_{n=1}^{N_m}u_n^i+\alpha_i^{(t)}-1}$$

- M Step:

$$\alpha^{(t+1)} = \arg\max_{\alpha}\sum_{m=1}^{M}\mathbf{E}_{q_m^{(t+1)}}[\log p(\mathbf{u}_m,\mathbf{f}_m,\theta^m)|\alpha,\beta]$$

The estimation of β is straightforward since it is decoupled from both α and θ during the M-step of the EM algorithm. Thus it does not require the E-step and can be solved once and for all. The MLE of β is the solution to the problem:

$$\begin{aligned} \max_{\beta} \quad & \sum_{m=1}^{M}\sum_{n=1}^{N_m}\sum_{i=1}^{K}\sum_{j=1}^{L}u_{nm}^i f_{nm}^j \log\beta_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^{L}\beta_{ij} = 1 \text{ for } i = 1,...,K. \end{aligned}$$

which is

$$\hat{\beta}_{ij} = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N_m}u_{nm}^i f_{nm}^j}{\sum_{m=1}^{M}\sum_{n=1}^{N_m}u_{nm}^i} \text{ for } \forall i,j$$

IV. Experimental Results

From the 11 week transaction trace, we use the procedures discussed in the previous section to estimate the Dirichlet parameter α and the conditional multinomial parameters β for each of the two file generation scenarios. The parameters used for the estimations and the subsequent simulations are the following:

- The number of periods: $M = 11$
- The number of causes: $K = 63$ for the LDA model; $K = 262$ with the observed user assumption of Section III-C2.
- The number of distinct files: $L = 2,123,734$
- The mean of the Poisson random variable N: $\xi = 471,659$

The choice of $K = 63$ for the LDA model is due to the observation of Section II-C where we clustered the users into 63 groups. For the observed user model, there were $K = 262$ distinct users.

Given the dimensionality of the parameters (see Section III-B), the resulting values cannot be shown explicitly. We note that all components of the estimated Dirichlet parameter were in the order of 10^{-4} for the LDA model and over 93% of the components were less than 1 for the model where users were assumed the cause. Since a small α indicates that the probability mass is concentrated in a few causes (or users), this is consistent with our observation of the real traces that the majority of the file access request is made by only a small number of users.

In order to validate the model, a synthetic trace is generated using the estimated parameters. We then conduct an analysis similar to the one of Section II, in order to check the statistical consistency between the real and synthetic traces.

Figure 7 plots the comparison of the resulting file popularity rank-frequency. We measured the goodness of fit of the real data for file popularity to the two generative models with the estimated parameters through a χ^2-test. It is possible to compute the marginal file access distribution of the model using the formula $p(f = j) = \sum_{i=1}^{K}\beta_{ij}\alpha_i/\sum_{k=1}^{K}\alpha_k$, which we use as the distribution under the null hypothesis. It is well known that χ^2-tests are problematic when there are a large number of bins where the expected frequency is small, because the χ^2 statistic tends to be inflated by the relatively small denominator. Since the observations of the tail part of our distribution is scarce, we suffer from this exact problem. But since we are primarily interested in the fit quality of the lower tail part, we truncated the data to ranks with expected frequency of at least 1. Considering that the general rule of thumb for the validity of χ^2-tests where at least 80% of the bins have expected frequency of at least 5, our truncation rule is very aggressive towards rejecting the null hypothesis. Nevertheless, in both models, we get a p-value of 1 and accept the null hypothesis.

For the model where observed users are the causes, we further study the statistic properties of the synthetic trace. We examined the rank-frequency plot for distinct files accessed

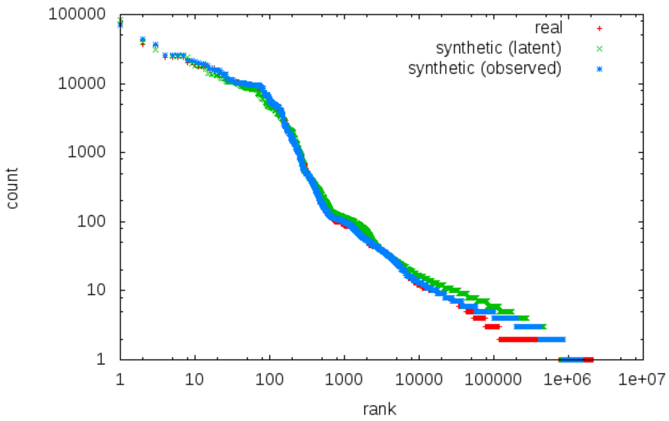


Fig. 7: File popularity, real and estimated with LDA.

plotted against user rank (Figure 8) and for the access count plotted against user rank (Figure 9). We observed that among the 5 top ranked users, who collectively account for more than 79% of the file accesses, the maximum error was 30.8% for Figure 8 and 26.6% for Figure 9 where error is evaluated by |count(real)-count(synthetic)|/count(real). Nevertheless, it is disturbing to observe an order of magnitude difference in the tail part. This discrepancy can be partially explained by recalling that our model only considers the set of users and files that appear at least once in the real traces during the 11 week period as the total population. This means the number of distinct users in the synthetic trace (216), by design, can never be larger than the number of distinct users in the real trace (262). Hence, the synthetic curve lies under the real curve in the tail region. To avoid this artifact, we could use a longer time period to fit the model in order to increase the universe of users and files, and then simulate logs only for a sub-period and compare it against the real traces of the corresponding period. However, the results as they are indicate that matching users alone with causes is not correct, illustrating the explanatory power of the proposed class of models.

V. Related Work

A first line of related work involving file access characterization derives from I/O workload analysis. Usually in these studies, the focus is on the access characteristic of individual applications [12], aggregate metrics such as I/O transfer size [13], and application specific I/O operations counts [14]. There has been early empirical studies in analyzing the distribution of file access operations on various commercial environments by providing simple statistics on the I/O operation counts [15].

Recently, with the emergence of large scale grid systems, more effort was put into characterizing the relationships between individual files and graph structures derived from these relationships. An important information comes from the comparison of our results with those of Doraimani and Iamnitchi [6], which analyzed trace data from the high energy physics experiment DZero [16]. Although the environment was similar in nature to ours, several of the results including the file popularity shared almost no similarity to our results. In their analysis, they do not profile the files according to who accessed it, but rather grouped files together according simultaneous access. Therefore, it is not possible to verify whether the discrepancy is due to the different behavior of the dominating users or because of a fundamental difference of the nature of the applications that run on the two different systems. However, the multi-disciplinary nature of our trace points to the last explanation.

A second line is related to predictive models. The focus is not on file access per se, but on bandwidth consumption and performance. [17] presents an extensive study on burst identification, modeling and prediction, in the context of e-science. The testbed, DQ2, a petabyte-scale distributed files management system for the Large Hadron Collider, is one of the largest existing open data system, with a mix of cloud and grid technologies. The approach is based on time-series, in the sense that no metadata are exploited, but only bandwidth measurements. The sobering conclusion is that all

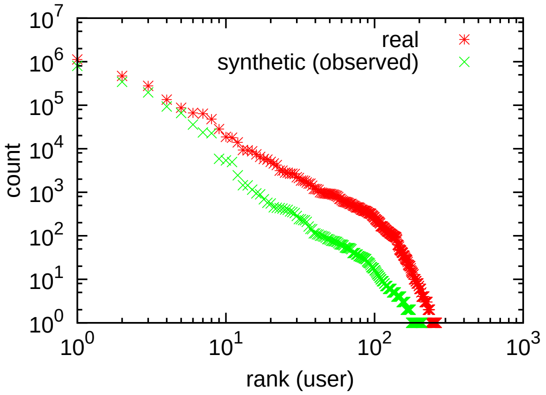


Fig. 8: Distinct files accessed for in the synthetic trace

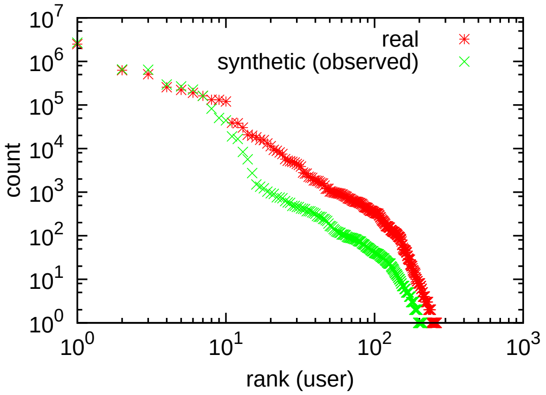


Fig. 9: Access frequency of users of synthetic trace

prediction methods (Neural Networks, SARIMA, and many others) essentially fail to predict bursts. This suggests that exploiting concepts of a higher semantic level, in the spirit of the latent causes that we described in the paper, might be a useful approach.

The literature about P2P systems, which are another style of community driven file management, mostly focus on social behavior (freeriding, seeding etc.), see for instance the recent descriptive analysis in [18]. An interesting conclusion of [18] is that the Bittorrent closed communities (those joined only by invitation), essentially behave as systems based on FTP transfers, with all data coming from the seeders. This fact opens the possibility to experiment our model in a very different context.

VI. Conclusion

Trace analysis of the LAL site revealed that most activity is initiated by a very small fraction of users, and consequently, the collective behavior is heavily influenced by them. Contrary to common belief, most frequency plots did not exhibit even local power law behavior. We conjecture that similar behavior should be present under similar regulation policies in other multi-disciplinary systems.

This raises a very challenging issue: moving to cloud technology might be they key for wider adoption of shared infrastructures by scientific communities other than HEP; but evaluating, dimensioning, and optimizing the future e-science infrastructures require an alternative to experimenting on real, large, and complex data. For instance, comparing our results with those of Doraimani and Iamnitchi shows very different behaviors. Well-founded and parsimonious representations, in other words generative models, are needed to experiment seamlessly, and may also contribute to the a priori knowledge required for operational autonomics. This paper proposed two generative models of the file access that characterize their spatio-temporal structure, and can be exploited in global simulation frameworks.

When observing the file access pattern from a workload analysis perspective, often the focus is on the application. In an e-science environment where applications are usually custom made, not much information is directly available to identify the type of the application. Therefore, we relied on LDA to define what the latent "applications" are and how they characterize the trace data, or assumed that users are synonymous with applications and therefore modeled the collective behavior of users.

Acknowledgment

This work has been partially supported by France Grilles and by the EGI-Inspire EC project FP7-RI-261323. The data have been provided by the Grid Observatory initiative www.grid-observatory.org.

References

[1] F. Gagliardi and al., "Building an Infrastructure for scientific Grid computing: status and goals of the EGEE project," *Philosophical Transactions of the Royal Society A*, vol. 1833, 2005.

[2] C. Germain-Renaud and al., "The grid observatory," in *11th IEEE/ACM Int. Symp. on Cluster Computing and the Grid*, 2011, pp. 114–123.

[3] S. Batra and S. Bawa, "Using lsi and its variants in text classification," in *Advanced Techniques in Computing Sciences and Software Engineering*, K. Elleithy, Ed. Springer Netherlands, 2010, pp. 313–316.

[4] D. Blei, Y. Ng, and M. Jordan, "Latent Dirichlet Allocation," *J. of Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[5] A. Clauset, C. R. Shalizi, and M. E. J. Newman, "Power-Law Distributions in Empirical Data," *SIAM REVIEW*, vol. 51, no. 4, pp. 661–703, DEC 2009.

[6] S. Doraimani and A. Iamnitchi, "File grouping for scientific data management: lessons from experimenting with real traces," in *Proceedings of the 17th Int. Symp. on High performance distributed computing*, ser. HPDC '08. New York, NY, USA: ACM, 2008, pp. 153–164.

[7] Anastasia Bezerianos and Fanny Chevalier and Pierre Dragicevic and Niklas Elmqvist and Jean-Daniel Fekete, "GraphDice: A System for Exploring Multivariate Social Networks," *Computer Graphics Forum - Eurographics/IEEE-VGTC Symposium on Visualization 2010 (EuroVis 2010)*, vol. 10, no. 3, pp. 863–872, June 2010.

[8] R. A. Davis, T. C. M. Lee, and G. A. Rodriguez-Yam, "Structural break estimation for nonstationary time series models," *J. Am. Stat. Assoc.*, vol. 101, pp. 223–239, March 2006.

[9] T. Elteto, C. Germain-Renaud, P. Bondon, and M. Sebag, "Towards non-stationary grid models," *J. Grid Computing*, vol. 9, no. 4, Dec. 2011, to appear.

[10] J. Dickey, "Multiple hypergeometric functions: Probabilistic interpretations and statistical uses," *J. Am. Stat. Assoc.*, vol. 78, pp. 628–637, 1987.

[11] T. Minka, "Estimating a Dirichlet distribution," M.I.T., Tech. Rep., 2000.

[12] N. Nieuwejaar, D. Kotz, A. Purakayastha, C. S. Ellis, and M. L. Best, "File-access characteristics of parallel scientific workloads," *IEEE T. Parall. Distr. Syst.*, vol. 7, pp. 1075–1089, October 1996.

[13] N. Nakka, A. Choudhary, W. K. Liao, L. Ward, R. Klundt, and M. I. Weston, "Detailed Analysis of I/O traces for large scale applications," in *16TH Int. Conf. on High Performance Computing, Proceedings*, 2009, Proceedings Paper.

[14] E. Smirni and D. Reed, "Lessons from characterizing the input/output behavior of parallel scientific applications," *Perform. Evaluation*, vol. 33, no. 1, pp. 27–44, JUN 1998.

[15] K. K. Ramakrishnan, P. Biswas, and R. Karedla, "Analysis of file i/o traces in commercial computing environments," *SIGMETRICS Perform. Eval. Rev.*, vol. 20, pp. 78–90, June 1992.

[16] The DZero Experiment, "http://www-d0.final.gov."

[17] M. Lassnig, T. Fahringer, V. Garonne, A. Molfetas, and M. Branco, "Identification, modelling and prediction of non-periodic bursts in workloads," in *10th IEEE/ACM Int. Symp. on Cluster Computing and the Grid*, 2010, pp. 485–494.

[18] M. Meulpolder, L. D'Acunto, M. Capotă, M. Wojciechowski, J. A. Pouwelse, D. H. J. Epema, and H. J. Sips, "Public and private bittorrent communities: a measurement study," in *Proceedings of the 9th Int. Conf. on Peer-to-peer systems*, ser. IPTPS'10. Berkeley, CA, USA: USENIX Association, 2010, pp. 10–10.