

A Novel and Generic Model for Synthesizing Disk I/O Traffic Based on The Alpha-stable Process

Qiang Zou^{*†}, Dan Feng^{*†✉}, Yifeng Zhu[‡], Hong Jiang[§], Xiaohu Ge^{*}, Zhihua Zhou^{*}
^{*} School of Computer, Huazhong University of Science and Technology, Wuhan, China
[†] Wuhan National Laboratory for Optoelectronics, China
[‡] University of Maine, Orono, ME, USA
[§] University of Nebraska-Lincoln, Lincoln, NE, USA
^{*} Huazhong University of Science and Technology, Wuhan, China
hustmathcs@gmail.com, dfeng@hust.edu.cn, zhu@eece.maine.edu, jiang@cse.unl.edu

Abstract

Obtaining representative and concise I/O workloads for the purpose of projecting the performance of storage systems remains a challenge due to the complex nature of I/O behaviors. Previous studies have shown that disk I/O traffic can be represented as an independent and identically distributed random process in some workloads and a self-similar process in others. Additionally, workloads in the presence of self-similarity can exhibit either Gaussian or non-Gaussian characteristics. This paper proposes a new and generic model based on the α -stable process to accurately build a synthetic workload representative of I/O traffic in production storage systems. The novelty of this new model is that it has the capability of characterizing both self-similar Gaussian and non-Gaussian workloads. Experimental results show that this model can accurately capture the complex I/O behaviors of real storage systems and more faithful than conventional models, particularly the burstiness and heavy-tail characteristics under the Gaussian and non-Gaussian workloads.

1. Introduction

Modeling I/O workloads accurately is one of the most fundamental aspects of performance evaluation in storage systems. However, I/O burstiness in workloads in production systems imposes significant challenges to this modeling process, thus making it difficult to accurately predict and effectively overcome the I/O performance bottleneck brought by I/O burstiness.

Previous studies on traces collected in production environments have shown a complex variety of I/O behaviors. In some environments, I/O inter-arrival times of requests

are uncorrelated to one another, and consequently the I/O arrival process can be modeled as independently and identically distributed (IID) [27], such as the Poisson process. In other environments, there are strong correlations in I/O inter-arrival times, and many researchers found that I/O activities in these workloads appear to be evidently self-similar [8, 9, 26]. Therefore, a versatile disk I/O workload model is needed to accurately synthesize the disk workload in both environments.

For I/O workloads with the presence of self-similarity, I/O burstiness is almost ubiquitous over their lifetime span. However, I/O burstiness in some workloads appears to be Gaussian, while I/O burstiness in some other workloads appears to be non-Gaussian. If the distinction between the Gaussian and non-Gaussian property is ignored in the synthesizing of workloads, it is difficult to truthfully and accurately represent the real degree of I/O burstiness in real workloads, thus causing potential mis-prediction of workloads. Therefore, a good model for I/O workload with self-similarity should not only be able to characterize the I/O bursty activities in Gaussian condition, but also in non-Gaussian condition. This paper proposes a stochastic model based on the α -stable process that meets this requirement.

Particularly, this paper analyzes the *search engine I/O* trace collected in 2002 [3] and three typical disk I/O traces (*tpc-d*, *cello96*, *cello99*) collected in different environments by HP Laboratories [1]. We examine the correlations of inter-arrival times in every workload, and study the Gaussian property of self-similar workload. Through distribution matching we find that I/O arrivals in each workload belong to the α -stable distribution. An I/O workload model based on the α -stable process is then presented. To the best of our knowledge, little research work has been done on this topic. This paper takes a first step toward using

an I/O workload model based on the α -stable process to synthesize I/O workloads. This paper makes the following conclusions.

1. Through studying the auto-correlation functions of each workload, we find that I/O arrivals are largely uncorrelated in *tpc-d* and *search engine I/O*, but significantly correlated in both *cello96* and *cello99*.
2. For those workloads with self-similarity, the Gaussian characteristic exhibits high variability. Interestingly, we find that *cello96* appears to be Gaussian while *cello99* is non-Gaussian. Thus it is important to select a traffic model capable of characterizing both the Gaussian and non-Gaussian property in order to accurately synthesize workload and obtain unbiased performance evaluation.
3. By using Quantile-Quantile plot and Probability-Probability plot, we find that I/O arrivals in all workloads studied in this paper follow the α -stable distribution. In addition, we quantify the parameters of the provided α -stable process by the maximum-likelihood estimate.
4. Based on the observation of the α -stable process, this paper develops an α -stable disk I/O workload model that can not only clearly represent the long-range dependence in self-similar disk traffics, but also accurately synthesize I/O workload with short-range dependence. This model is a powerful and versatile stochastic tool to generate synthetic I/O workload in which I/O arrivals can be either independent identically distributed, Gaussian self-similar, and non-Gaussian self-similar. Our experimental results show that this model is accurate and more faithful than conventional models.

The rest of this paper is organized as follows. Section 2 gives an overview of the related work. Section 3 presents our motivations for using the α -stable process to model disk traffics. Section 4 then presents the α -stable disk I/O workload model. Section 5 describes the rationality of using α -stable distribution with real traces. Section 6 compares the workloads synthesized by the proposed model with real traces and others synthetic workloads. Section 7 concludes this paper.

2. Related Work

Two basic approaches are widely used to obtain representative I/O workloads. One is to collect I/O traces in a production environment and then carefully replay the traces during simulation or experiment [24]. The other is to use

synthetic I/O requests that emulate the behaviors of actual workloads [4]. The second approach allows us to flexibly study the effects of some workload parameters [12].

Prior works in the literature have shown that the data traffic burstiness in many workloads exhibits the property of self-similarity, such as the multimedia traffic [19], Ethernet [13] and web [5] traffic. Various self-similar models have been proposed to emulate the burstiness, but they all have limitations. For example, FARIMA [7] is proposed to generate synthetic Variable Bit Rate (VBR) video traces but it is not intrinsically bursty. The FBM model used by several researchers [12, 16, 22] is easy to construct and can model the self-similarity under the Gaussian condition. However, the FBM model cannot represent both the long-range and short-range dependence simultaneously.

To emulate the burstiness in storage systems, models that aggregate multiple ON/OFF sources have been proposed [8–10]. Specifically, Gomez et al. [8, 9] demonstrate that disk-level I/O requests are self-similar in nature, and perform a structural modeling showing that the self-similar behavior can be explained by combining two different approaches: an ON/OFF source model and Cox’s model [18]. Gribble et al. [10] find that sources of short-term file system traces exhibit ON/OFF behavior and proposes a simple technique for synthesizing a stream of events that exhibit the same self-similar short-term behaviors as was observed in the file system traces. The ON/OFF model is easy to construct, and its parameters have physical meanings. However, the ON/OFF model only adapts to synthesize particular classes of traffic and constructing the model often requires significant computational effort [26].

Wang et al. [25] proposed a two-dimensional model that characterizes both the temporal and spatial behaviors of data accesses. In addition, by using the joint entropy of the two-dimensional disk request arrival events (time and space), this model can model the spatio-temporal correlations. However, the Gaussian property characterized by this model belongs to the spatial domain, not the temporal domain. Wang et al. [26] further propose to use the b -model to model bursty disk traffic. The model only depends upon a single parameter, bias p , which can be estimated from real traces. However, Hong et al. [11] claim that it may not be realistic to monitor any disk workload and model it accurately with some small number of parameters. Hong et al. [11] then use binomial multifractal to generate synthetic arrival patterns at relatively small time scales, improving the accuracy of trace generation, but not to synthesize workload at long time scales. In general, it is challenging to accurately model I/O workloads for traffic prediction and performance evaluation in storage systems.

Based on the α -stable process, this paper proposes an I/O workload model to generate synthetic disk I/O requests, with a focus on precisely emulating the burstiness that is

often observed in real systems. Our model is flexible to represent both long-range dependence and short-range dependence. In addition, this model can accurately characterize the I/O burstiness in disk workloads under the Gaussian conditions as well as the non-Gaussian conditions.

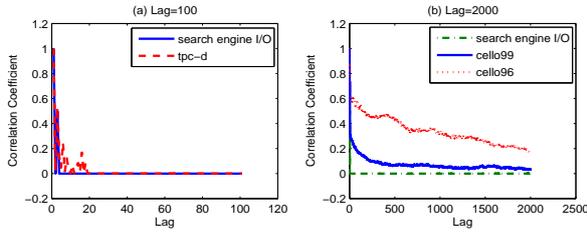


Figure 1. Auto-correlation functions (ACFs) of I/O request inter-arrival times for *search engine I/O*, *tpc-d*, *cello96* and *cello99*, respectively.

3. Why The α -stable Process?

Burstiness in disk I/O arrivals exists prevalently at all time scales [8]. And one of the main characteristics in the α -stable process is its ability to represent precisely the burstiness in stochastic phenomenon. That directly motivates us to develop a new model based on the α -stable process.

This section studies the auto-correlation functions (ACF) of inter-arrival times [6, 27], and shows the correlations of I/O inter-arrival times in *search engine I/O*, *tpc-d* and *cello* workloads. ACF is a widely-used mathematical tool to study the correlations, i.e., measure if earlier values in a time sequence $X = \{X_i | i = 1, \dots, N\}$ have some relation to later values. The correlation coefficient at lag k is defined as

$$c_k = \frac{1}{N-K} \sum_{i=1}^{N-k} (X_i - \bar{X})(X_{i+k} - \bar{X}) \quad (1)$$

where \bar{X} is the expectation of the time series X . Then the ACF(k), with a lag of k is

$$ACF(k) = \frac{c_k}{c_0} \quad (2)$$

As shown in Fig. 1(a), there is a slight correlation for I/O inter-arrival times in the *tpc-d* and *search engine I/O* workloads only for a very short-term time scales ($lag=20$ for *tpc-d* and $lag=5$ for *search engine I/O*). However, both are largely not correlated in longer time scales. Taking the *search engine I/O* workload for an example in Fig. 1(b), nearly all of the correlation coefficients of I/O inter-arrival time are equal to zero. This indicates there is little correlation for I/O inter-arrival time in long-term time scales

($lag=2000$). As shown in Fig. 1(b), the correlation of I/O inter-arrival times in the *cello* workload is evident in long-term time scales ($lag=2000$), especially for *cello96*. This shows that the correlations of inter-arrival times in different workloads are different with one another. The α -stable process is just the right model to adequately represent the short-range and long-range dependence.

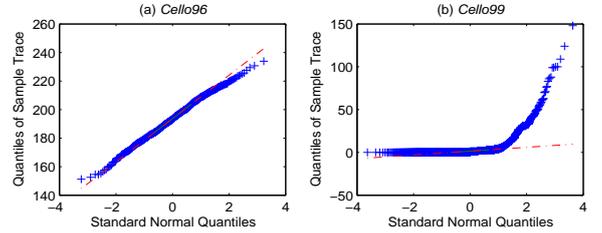


Figure 2. Examine the Gaussian property of *cello96* and *cello99* workloads through QQ plots of sample data versus standard normal.

Prior studies have concluded that the *cello* workloads are self-similar in nature [8,9,11]. In self-similar I/O workloads, there are long-range dependence and heavy-tail distribution characteristics, both of which render the variance of I/O arrival stochastic process infinite. In addition, the maximum of I/O inter-arrival times in clients' requests approaches infinity theoretically, which also makes the variance unbounded. According to the generalized central limit theorem (GCLT), the normalized aggregation of infinitely independent and identically distributed sources with infinite variance converges to a family of α -stable marginal distributions if its marginal distributions converge. Consequently, using the α -stable process to model disk I/O workloads becomes a nature choice.

For the self-similar *cello* traces, by using the normal quantile plots (QQ plots) to measure the Gaussian property [22], as shown in Fig. 2, we find that *cello96* is Gaussian and *cello99* is non-Gaussian. In Fig. 2(a), all of the scatter points corresponding to an I/O arrival event given in the traces evidently follow a straight line, indicating that *cello96* is Gaussian. In Fig. 2(b), all of the scatter points evidently don't fall into a straight line but an increasing curve, indicating that *cello99* is non-Gaussian. The results above show interestingly that some self-similar I/O workloads have the Gaussian property while others do not. Therefore, the model used to capture the I/O burstiness in *cello* should be able to represent both the Gaussian and non-Gaussian properties. The α -stable process can not only meet this requirement, but also generate synthetic I/Os that better emulate the I/O burstiness seen in real systems, as demonstrated later.

In summary, the α -stable process has a solid theoretical basis for synthesizing disk I/O workloads. This motivates us to explore the α -stable process in I/O modeling, with an aim to improve our understanding of disk I/O burstiness characteristics and design a better synthetic trace generating tool for performance evaluation of storage systems.

4. The α -stable Disk I/O Workload Model

In most stable distributions, densities and distribution functions are not known in closed forms. Thus, in general α -stable distributions are defined by their characteristic functions.

Definition 4.1 A random variable X is said to have a stable distribution if there exist parameters $0 < \alpha \leq 2$, $\sigma > 0$, $-1 \leq \beta \leq 1$, and $\mu \in \mathbb{R}$ such that its characteristic function has the following form [21]:

$$E[e^{i\theta X}] = \begin{cases} e^{-\sigma^\alpha |\theta|^\alpha (1 - i\beta \text{sign}\theta \tan \frac{\pi\alpha}{2}) + i\mu\theta}, & \alpha \neq 1 \\ e^{-\sigma |\theta| (1 + i\beta \text{sign}\theta \ln|\theta|) + i\mu\theta}, & \alpha = 1 \end{cases} \quad (3)$$

where $\text{sign}\theta = \begin{cases} 1, & \theta > 0 \\ 0, & \theta = 0 \\ -1, & \theta < 0 \end{cases}$, α , β , σ and μ are characteristic exponent, skewness parameter, scale and location parameters, respectively.

The characteristic exponent α represents the level of burstiness in the distribution. The distribution can be skewed if the skewness parameter β is different from zero. Variables σ and μ are called the scale and location parameters, respectively, and represent the deviation and the mean of the distribution. A random variable X that follows an α -stable distribution with the above parameters is denoted by $X \sim S_{\sigma,\beta,\mu}^\alpha$ [21].

If $0 < \alpha < 2$, the characteristic function of α -stable distribution is one class of non-Gaussian functions. If $\alpha = 2$, the characteristic function will degenerate to a Gaussian one, shown as $E[e^{i\theta X}] = \exp\{-\sigma^2 \theta^2 + i\mu\theta\}$. In fact, this is the characteristic function of a Gaussian stochastic process, with a constant mean μ , and variance $2\sigma^2$, β becomes of no meaning due to $\beta \tan \pi = 0$, according to the characteristic function of α -stable process. Therefore, by changing α , the α -stable process is able to represent the stochastic process under Gaussian condition as well as non-Gaussian condition.

The stochastic process studied in this paper is a class of the α -stable process that has both the self-similarity and the stable increments. Extending FBM under the α -stable condition, we can obtain various forms, of which one is the Linear Fractional Stable Motion (LFSM) [20]. LFSM shares all properties of the α -stable process, and its increment process is called the Linear Fractional Stable Noise

(LFSN). The LFSN process can be expressed in a discrete domain, which makes it one of the most common mathematical modeling tools [23].

Due to the fact that realistic modeling exists usually in discrete states, the LFSN process expression in continuous states above needs to be transformed to a discrete expression, replacing the integral with a *sum* function. Through the discrete transformation of the properties of the α -stable process [21], we can express a LFSN process as a linear convolution as follows:

$$\begin{aligned} N_{\alpha,\beta,H}(i) &= (h_d \cdot S_{1,\beta,0}^\alpha)(i) \\ &= \sum_{k=1}^{Km} h_d\left(\frac{k}{m}\right) \cdot S_{\left(\frac{1+\beta}{2}\right)^{\frac{1}{\alpha}},1,0}^\alpha\left(i - \frac{k}{m}\right) \\ &\quad - \sum_{k=1}^{Km} h_d\left(\frac{k}{m}\right) \cdot \tilde{S}_{\left(\frac{1-\beta}{2}\right)^{\frac{1}{\alpha}},1,0}^\alpha\left(i - \frac{k}{m}\right) \end{aligned} \quad (4)$$

where $h_d(x) = \begin{cases} x^d - (x-1)^d, & x \geq 1 \\ x^d, & 0 < x \leq 1 \end{cases}$, $d = H - \frac{1}{\alpha}$,

$S(i)$ is an α -stable stochastic variable that is independently and identically distributed, h_d is the discrete inner-kernel function, m is the grid parameter in the integral-discretizing scheme, K is the integral stop point. $N_{\alpha,\beta,H}(i)$ represents the discrete form of the stable LFSN process (i.e. a class of the α -stable process satisfying $\sigma = 1, \mu = 0$), $S_{1,1,0}^\alpha$ and $\tilde{S}_{1,1,0}^\alpha$ represent two independently and identically distributed discrete stochastic variables, the common distribution is $S_{1,1,0}^\alpha$. H is the Hurst parameter, and gives a measure of the degree of self-similarity of a given time-series, $0 < H < 1$. The Hurst parameter to a set of observations can be estimated by R/S analysis (i.e., *Pox plot*), and a detailed description of this method can be found in [5, 7, 13].

Since the marginal distribution of a LFSN process is an α -stable process, the LFSN process has the basic properties of the α -stable process. This paper provides a novel model directly based on the LFSN process theory. According to the properties of the α -stable process, we construct an α -stable I/O workload model, and its formalization is expressed as follows:

$$IOs(i) = v \cdot N_{\alpha,\beta,H}(i) + \delta \quad (5)$$

where $IOs(i)$ represents the number of I/O requests in the i^{th} unit time, v and δ are real numbers above zero.

This model includes five parameters, and the physical meaning of each parameter is given as: α measures the degree of I/O burstiness, β represents the degree of heavy tail in the I/O traffic, H measures the degree of self-similarity, v represents the I/O mean velocity of the disk traffic, and δ represents the deviation degree relative to I/O mean velocity of disk traffic.

In the following section we will analyze realistic I/O traces, and then carefully scrutinize the rationality of adopting the α -stable process with credible experimental data.

Table 1. Estimates the parameter of α -stable distribution based on maximum-likelihood estimate.

| Data Set | Search Engine I/O Sample Trace | α | β | σ | μ |
|----------|--------------------------------|----------|---------|------------|---------|
| 1 | WebSearch1 | 2.0000 | \ | 55.0372 | 332.000 |
| 2 | WebSearch2 | 2.0000 | \ | 58.1822 | 301.000 |
| 3 | WebSearch3 | 1.7284 | -0.9004 | 55.9710 | 184.168 |
| Data Set | Tpc-d Sample Trace | α | β | σ | μ |
| 4 | Q2 | 1.3377 | 1.0000 | 100.365 | 456.644 |
| 5 | Q3 | 0.7838 | 0.1142 | 61.4298 | 980.686 |
| 6 | Q4 | 1.8810 | -1.0000 | 107.589 | 1937.13 |
| 7 | Q6 | 0.6950 | -0.6929 | 51.4305 | 2114.49 |
| 8 | Q8 | 1.6504 | 1.0000 | 1159.70 | 1197.01 |
| Data Set | Cello96 Sample Trace | α | β | σ | μ |
| 9 | 10-15-09 | 2.0000 | \ | 38.7881 | 2.0000 |
| 10 | 10-28-11 | 2.0000 | \ | 30.9257 | 8.0000 |
| 11 | 12-02-16 | 2.0000 | \ | 60.2789 | 77.0000 |
| 12 | 12-02-17 | 2.0000 | \ | 76.5279 | 60.0000 |
| 13 | 12-02-20 | 2.0000 | \ | 59.7547 | 18.0000 |
| Data Set | Cello99 Sample Trace | α | β | σ | μ |
| 14 | 04-16-11 | 1.5973 | 1.0000 | 11.4362 | 15.2522 |
| 15 | 05-11-21 | 0.8398 | 1.0000 | 0.8223 | -0.4903 |
| 16 | 06-29-13 | 1.1961 | 1.0000 | 4.3797 | 0.8807 |
| 17 | 09-10-12 | 0.4018 | 1.0000 | 0.1832E-01 | -0.4786 |
| 18 | 12-31-11 | 0.8295 | 1.0000 | 0.4893 | -1.3384 |

After that we will use real I/O traces to examine whether real trace data follow the α -stable distribution.

5. Examination of α -stable Distribution

To examine whether I/O arrivals specified in an I/O trace follow the α -stable distribution, we first estimate the parameters of a given α -stable distribution by measuring the dataset, then compare the estimated distribution against the real distribution of the I/O traces. There are various mathematical methods to estimate the parameters of an α -stable distribution. In this section, three typical methods are chosen, including the maximum-likelihood estimate, the quantile method and the sample characteristic function [15]. In addition, two different statistical analysis methods are used to compare the estimated and real distributions. These different methods can compensate one another and reduce the estimation errors.

5.1. Estimation of the Parameters

Search engine I/O includes three I/O traces from a popular search engine, i.e., *WebSearch1*, *WebSearch2* and *WebSearch3* [3]. There are 17 queries in *tpc-d*, and these queries last different amounts of time. While 46 data sets in *cello96* were collected during one and half months, the data sets in *cello99* spread over a time period of approximately one year. Both *cello96* and *cello99* consist of 24

subtraces, with each one lasting one hour. We compute the number of I/O arrivals per second. For each workload, we randomly select three or five data sets.

First, the maximum-likelihood estimation method is used to estimate the parameter values of the α -stable distribution provided for these 18 data sets. In Table 1, dates for the *cello* traces are in the MM-DD-HH format, and each row includes the sequence number of data set, sample data set and the estimates of four parameters. Due to the fact that the α -stable distribution will degenerate to a Gaussian stochastic process if $\alpha = 2$, with mean value μ and variance $2\sigma^2$, β will be meaningless (see section 4). Accordingly in Table 1, a slash will fill in the place of β if $\alpha = 2$. This measurement results prove the validity of Fig. 2(a) again, i.e., the *cello96* workload is Gaussian. According to the parameter estimates, we can obtain the relevant α -stable distribution and further check whether the given α -stable distribution is consistent with the I/O arrival process in real workloads. Due to the space constraint, this paper only discusses the results of the first, sixth, twelfth and fourteenth data set from Table 1.

5.2. Distribution Test

Based on a given α -stable distribution, through QQ (Quantile-Quantile) plots [14] we can judge whether the given α -stable distribution matches the probability distribution of given data set.

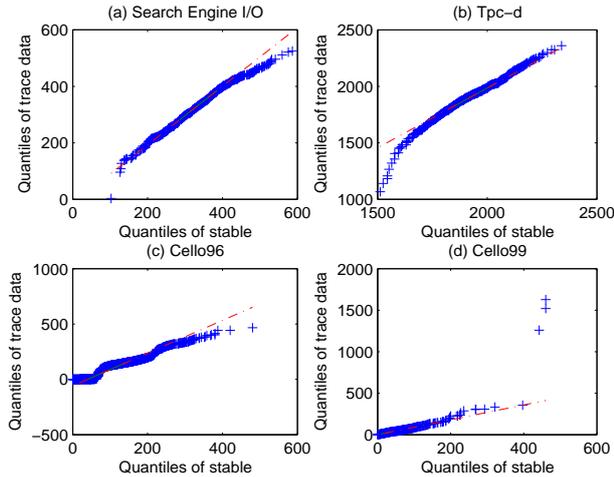


Figure 3. QQ plots of *search engine I/O*, *tpc-d*, *cello96* and *cello99* sample data versus α -stable distribution, respectively.

In order to examine the matching degree between the α -stable distribution and real data, the QQ plots of the given data set and α -stable distribution are illustrated in Fig. 3. As shown in Fig. 3, the X-axis shows the quantile value of the hypothetical α -stable distribution, and the Y-axis denotes the quantile value of the given data set. Fig. 3(a)-(d) illustrate the matching results corresponding to the selected four data sets. Through analyzing the QQ plots, we find that a majority of data points line along an approximate straight line. Therefore, it is reasonable to consider that the hypothetical α -stable distribution is consistent with the real data distribution.

While QQ plot is an effective tool, it has some limitations. First of all, as shown in Fig. 3, most of the points are compressed in a very narrow range in both dimensions, which limits our visual observation directly. Secondly, the tail of data points often fluctuate around the theoretical straight line and even beyond the target straight line area in Fig. 3(c), which is induced by the accumulative effect brought by the heavy-tail distribution.

In order to overcome the limitations of QQ plot, PP (Probability-Probability) plot [14] is deployed. As shown in Fig. 4, the X-axis shows the probability point of the hypothetical α -stable distribution, and the Y-axis denotes the probability point of the given data set. Instead of the quantile points in QQ plot by the probability point, PP plot is able to avoid the accumulative effect brought by the heavy-tail distribution. In Fig. 4, a majority of data points are located on a line of 45 degrees, i.e., the hypothetical α -stable distribution is consistent with the real data distribution.

In summary, the two different methods presented above give the same conclusion that the hypothetical α -

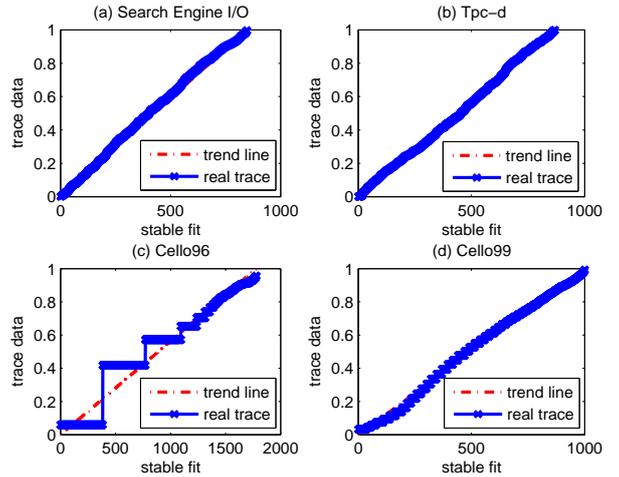


Figure 4. PP plots of *search engine I/O*, *tpc-d*, *cello96* and *cello99* sample data versus α -stable distribution, respectively.

stable distribution is consistent with the real data distribution. In addition, we have also used the quantile method and the sample characteristic function instead of the maximum-likelihood estimation, to perform all experiments conducted above. The results are consistent with the maximum-likelihood estimation. Thus, we conclude that I/O arrivals in disk I/O workloads can be modeled as an α -stable distribution.

6. Synthesizing the Disk Traffics

For *cello99*, the subtrace in each day includes 24 data subsets. We take a subset as a testing unit and accordingly each subtrace is divided into 24 testing units. For each testing unit, we compute the I/O arrival rate, i.e., the number of I/O arrivals per second. We put all arrival rates in a group of stochastic numbers. The maximum-likelihood method is used to estimate the parameters of the α -stable process corresponding to the data sets needed to be measured. And the relevant I/O workload can be synthesized corresponding to the data sets based on the constructed model. Then we are able to synthesize the I/O workload using the α -stable workload model.

The algorithm for synthesizing workload by the α -stable disk I/O workload model is given below.

ALPHA-STABLE-SYNTHETIC-TRACE-GENERATION

INPUT: I/O mean velocity v , the deviation degree of I/O mean velocity δ , grid parameter m , integral stop point K , original trace data file f .

OUTPUT: an access series ($IOs(1), IOs(2), \dots, IOs(n)$).

ALGORITHM:

```

for each  $f$ 
  //ALPHA-STABLE-PARAMETER-ESTIMATE
  Use maximum-likelihood estimate to estimate the
  parameter value  $\alpha$ ,  $\beta$ ,  $\sigma$  and  $\mu$  of the given  $\alpha$ -stable
  distribution for data sets in  $f$ ;
  if  $\alpha \notin (0, 2]$  or  $\beta \notin [-1, 1]$  or  $\sigma \leq 0$ 
    then break;
  else
    use Pox plot to estimate the Hurst value  $H$ 
    if  $H \notin (0, 1)$  or  $H = 1/\alpha$ 
      then break;
    else calculate  $d = H - 1/\alpha$ , and then  $h_d(x)$ 
      Set the initial values of  $m$  and  $K$ , and obtain
       $\{N_{\alpha, \beta, H}(i) : i = 1, 2, \dots, n\}$  using Equation (4)
      Set the initial value of  $\nu$  and  $\delta$ , and obtain
       $\{IOs(i) : i = 1, 2, \dots, n\}$  using Equation (5)
end for

```

In order to check the effect of disk I/O workload model based on the α -stable process, this section takes *search engine I/O*, *tpc-d*, *cello96* and *cello99* as targets to synthesize I/O workload, respectively. The synthetic workload will be compared with real disk traffics and the workload synthesized by the conventional traffic models.

Table 2. The trimmed mean of errors.

| Data Set | α -stable | Normal | Poisson | Lognorm |
|--------------------------|------------------|--------|---------|---------|
| <i>Search Engine I/O</i> | | | | |
| 1 | 2.47 | 3.16 | 6.43 | 3.21 |
| 2 | 2.79 | 1.49 | 2.32 | 5.02 |
| 3 | 0.28 | 4.17 | 7.45 | \ |
| <i>Tpc-d</i> | | | | |
| 4 | 24.26 | 34.94 | 47.10 | \ |
| 5 | 14.84 | 25.08 | 99.24 | \ |
| 6 | 5.25 | 21.66 | 26.30 | 48.35 |
| 7 | 6.43 | 157.29 | 233.79 | 183.09 |
| 8 | 5.98 | 31.51 | 645.45 | \ |

6.1. Analysis of Errors

According to the 18 data sets in Table 1, we use the proposed model and conventional models to synthesize the various workloads one by one. Because a badly skewed datum in a data set can potentially render the mean of the set arbitrarily skewed from the centers of the remaining data in the set, *the trimmed mean* [14] is used to evaluate the matching degrees between each real workload and the corresponding synthetic workloads. The trimmed mean of a data set is the arithmetic mean after trimming a small portion off each of the two ends of the sample data, making it

Table 3. The trimmed mean of errors.

| Data Set | α -stable | FBM | FARIMA | ON/OFF |
|----------------|------------------|--------|--------|--------|
| <i>Cello96</i> | | | | |
| 9 | 12.82 | 2.94 | 15.92 | 16.22 |
| 10 | 21.02 | 17.48 | 17.17 | 17.73 |
| 11 | 8.88 | 112.01 | 61.22 | 3.02 |
| 12 | 18.00 | 75.00 | 61.18 | 20.04 |
| 13 | 41.83 | 96.44 | 38.71 | 24.98 |
| <i>Cello99</i> | | | | |
| 14 | 0.41 | 126.68 | 20.24 | 5.60 |
| 15 | 1.45 | 49.71 | 1.15 | 5.07 |
| 16 | 3.28 | 187.13 | 5.61 | 7.25 |
| 17 | 0.69 | 74.92 | 0.45 | 3.15 |
| 18 | 1.76 | 75.67 | 0.85 | 7.26 |

more stable and resilient to abnormal data than the conventional average of samples expectation such as the arithmetic mean.

The trimmed mean of errors are illustrated in Table 2 and Table 3. First, we use the α -stable, Normal, Poisson, and Lognorm methods to synthesize the workloads corresponding to the *search engine I/O* and *tpc-d* traces, respectively. As shown in Table 2, a slash means that the Lognorm method is not able to synthesize workload due to the fact that in the relevant workloads the number of I/O arrivals in a resolution is zero. As can be seen from Table 2, generally the trimmed mean of error between the real workload and the α -stable synthetic workload is minimum, with the exception in which the trimmed mean of error between the 2nd data set and the α -stable synthetic workload is 2.79. This is understandable since the α -stable model is developed to capture the essence of all workloads synthetically way, not any one specific workload. Nevertheless, the matching degree of the α -stable synthetic workload for the 2nd data set is still reasonably good.

Next, we use the α -stable, FBM, FARIMA and ON/OFF models to synthesize the workloads corresponding to *cello96* and *cello99* traces, respectively. As shown in Table 3, for *cello99*, the trimmed mean of error between the real workload and the α -stable synthetic workload is minimum. And for some of the data sets in *cello96*, the trimmed mean of error between the real workload and the α -stable synthetic workload is minimum or close to minimum, e.g., the 9th, 11th and 12th data sets. In addition, the trimmed mean of error for the α -stable synthetic workload corresponding to the 13th data set is close to the maximum, especially the trimmed mean of error for the α -stable synthetic workload corresponding to the 10th data set. However, either for the 13th data set or the 10th data set, the error is only slightly larger than the corresponding minimum error. Especially for the 10th data set, the trimmed mean of error for the α -stable synthetic workload is larger than others, but only by a margin of 3.85 over the minimum error, indicating that

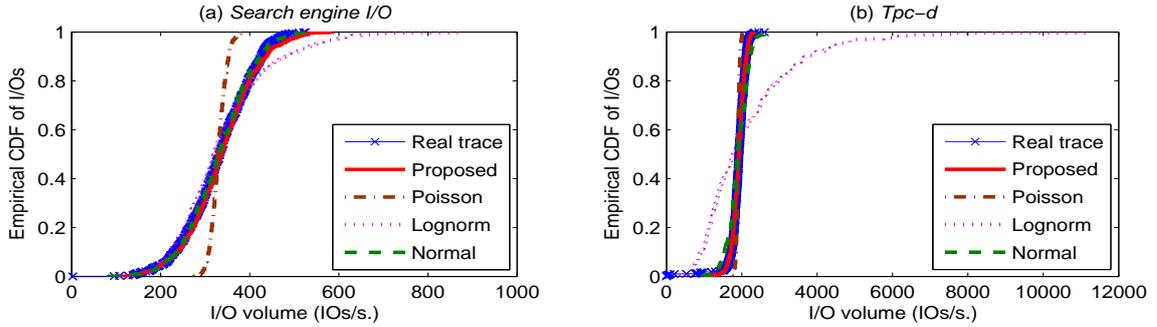


Figure 5. CDFs of synthetic I/O traces and real trace for *search engine I/O* and *tpc-d*.

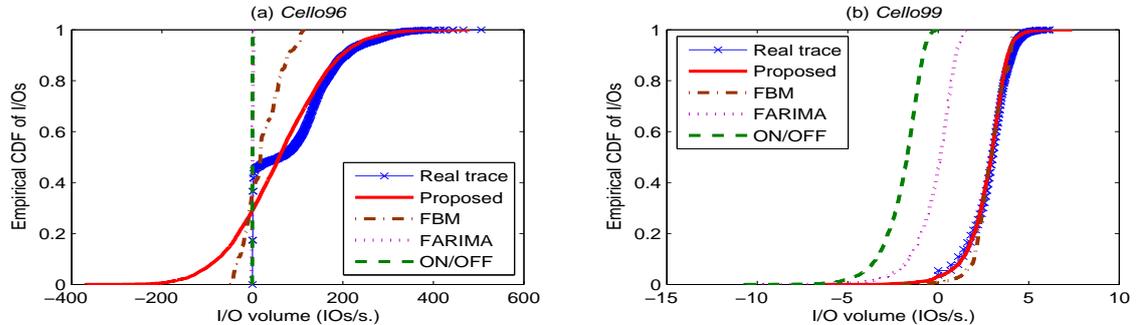


Figure 6. CDFs of synthetic I/O traces and real trace for *cello96* and *cello99*.

the matching degree between the real workload and the α -stable synthetic workload is still reasonably good.

6.2. Empirical Study

In order to intuitively present the synthetic workloads and comparative results, without the loss of generality, we select one group of the synthetic workloads from the *search engine I/O*, *tpc-d*, *cello96* and *cello99*, respectively. The *cumulative distribution functions* (CDFs) of the selected workloads, the 1st, 6th, 12th and 14th data sets, are illustrated in Fig. 5 and Fig. 6, the X-axis shows the I/O arrival numbers per second (thereinto, the X-axis in Fig. 6(b) shows the logscale), and the Y-axis denotes the percentage of the number of I/O arrivals. A point $(x; y)$ in the cumulative distribution curve indicates that $y\%$ of arrival rates are less than or equal to an arrival rate of x .

As can be seen from Fig. 5, the disk I/O workload synthesized by the α -stable model very closely matches the real trace data, especially for *tpc-d*. A quantitative approach to evaluate the improvement is to analyze the error. Taking the I/O workload synthesized by the Poisson model for an example, using the Maximum-likelihood estimate method to obtain the parameter value of Poisson distribution provided for those data sets in *search engine I/O* and *tpc-d*, we get the values of 27.1, 1.8999e+003, respectively. For *search engine I/O* and *tpc-d*, the trimmed means of errors between the real data set and the synthesized workload

through the Poisson model with these parameter values are 6.43, and 26.3, respectively, and the trimmed means of errors between the real data set and the synthesized workload through our proposed model are 2.47, and 5.25, respectively. Accordingly, our proposed model can reduce the trimmed mean of error of the Poisson models by 61.6%, and 80%, respectively.

As can be seen from Fig. 6, the disk I/O workload synthesized by the α -stable model very closely matches the real trace data, especially for *cello99*. Taking the I/O workload synthesized by the ON/OFF model for an example, for *cello96* and *cello99*, the trimmed means of errors between the real data set and the synthesized workload through the ON/OFF model are 20.04, and 5.6, respectively, and the trimmed means of errors between the real data set and the synthesized workload through our proposed model is 18, and 0.41, respectively. Accordingly, our proposed model can reduce the trimmed mean of error of the ON/OFF model by 10%, and 92.6%, respectively.

6.3. Discussion

The comparative analysis above shows that the α -stable I/O workload model is able to synthesize various disk-level I/O workloads accurately and flexibly. However, besides the disk-level I/O workload, there are still other various classes of I/O workloads such as file-level I/O workloads studied in Ref. [10], I/O workloads for scientific applica-

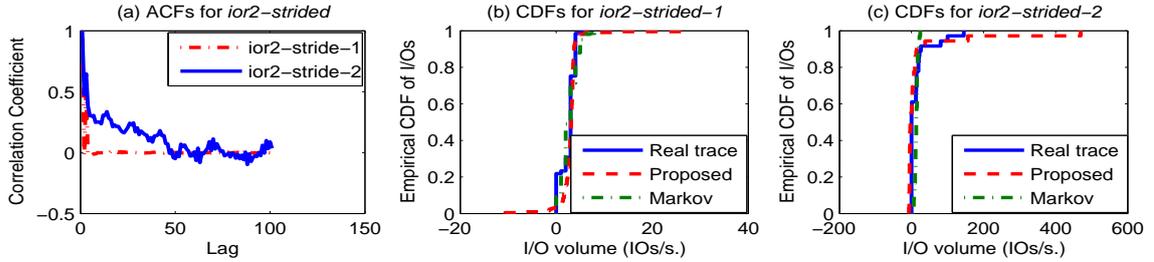


Figure 7. From (a) to (c): plot (a) illustrates the ACFs of I/O request inter-arrival times for *ior2-strided*. Plot (b) and (c) show the CDFs of synthetic I/O traces and real trace for *ior2-strided*, respectively.

tions studied in Ref. [17], etc.. Then whether the α -stable I/O workload model is also able to synthesize file-level I/O workloads and characterize the I/O behavior of scientific applications as accurately as it does for the I/O burstiness in the disk-level I/O workload remains unknown.

In our preliminary study of the parallel I/O workloads in the 2003 LLNL trace (including three scientific applications: *ior2*, *fl* and *m1*, and detailed description of these applications can be found in Ref. [2].) collected in a large-scale scientific computing environment, we find that the correlations of inter-arrival times in the I/O workloads for some of the application nodes are evident but unnoticeable in others. Taking *ior2-strided* (a subtrace of *ior2*) for an example, the correlations of inter-arrival times in the I/O workloads for some application nodes are strong, while there is little correlation of inter-arrival times in some others, as shown in Fig. 7(a). This result indicates that the arrivals series in some nodes are independently and identically distributed but not in all nodes. Thus it is reasonable to doubt the rationality of using the Markov model to model and predict I/O requests for scientific applications [17].

In this case, a generic traffic model is necessary to effectively synthesize these I/O workloads in scientific applications. This paper uses the α -stable IO workload model to synthesize the *ior2* I/O workload and compares the workload synthesized by our proposed model with the Markov model in Fig. 7(b)-(c). In Fig. 7(b), the trimmed means of errors between the real data set and the synthesized workload through the Poisson and our proposed models are 1.3, and 1.56, respectively. As shown in Fig. 7(b), the matching degree of our proposed model is comparable to the Markov model, due to the fact that the I/O arrival process is independently and identically distributed and I/O arrival process can be modeled by the Markov model as described in Ref. [17]. However, as can be seen from Fig. 7(c), the matching degree of our proposed model is better than the Markov model. In Fig. 7(c), the trimmed means of errors between the real data set and the synthesized workload through the Poisson and our proposed models are 22, and 14.9, respectively. For the workload synthesized by the Markov model, an evident deficiency is that it is diffi-

cult for the Markov model to capture the burstiness in I/O workloads. This indicates that the α -stable I/O workload model is very likely to be useful in synthesizing the I/O workload in the scientific computing environment, and our proposed model has good potential to be explored and exploited to provide useful insight into the synthesis of various I/O workloads.

7. Conclusions

The ability to accurately model disk I/O workloads mathematically is important and valuable to systems designers and practitioners since it can help reveal intrinsic relationships between the I/O traffic characteristics and the basic phenomenon in I/O workload (e.g., I/O burstiness) and provide the theoretical basis to predict I/O workloads, thus facilitating and guiding possible design optimizations, such as resource distribution and bottleneck identification and removal, to improve I/O performance and assure QoS. Unfortunately, accurately modeling disk I/O workloads remains a challenging issue due to the Burstiness in the arrival process. This paper proposed and evaluated a novel and generic mathematical model, the α -stable disk I/O workload model, to accurately synthesize I/O workloads.

Through studying the correlations of I/O inter-arrival times in some representative disk workloads, examining the Gaussian property of self-similar workloads and comparing with other I/O workload model, and simulation experiments, we have shown that our α -stable disk I/O workload can accurately describe the I/O arrival process with the property of either independent identically distribution, or Gaussian self-similarity, or non-Gaussian self-similarity. We compare our model against conventional models, including the Normal, Poisson and Lognorm methods for the IID workloads, and the FBM, FARIMA, ON/OFF for self-similar workloads. Experiment results show that the synthetic traces generated by our model can more faithfully emulate the I/O bursty arrival behaviors than the other methods. Specifically our model can more generally describe the heavy-tail characteristic in self-similar I/O traf-

tics. In addition, our model has five input parameters and each one has its physical meaning, allowing us to conveniently turn the I/O workload model for different environments. For example, we can change the values of parameter α so that our model is flexible to characterize the burstiness in disk I/O workload under both the Gaussian and non-Gaussian conditions.

8. Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments in reviewing this paper. We also thank HP and Umass trace repository for providing accesses to various tools and traces. Thanks go also to Lei Tian, Zhidong Wang, Li Li and Chujin Li for their help in writing this paper. This work is supported by the National Basic Research Program of China (973 Program) under Grant No.2004CB318201, the China NSF under Grant No.60503059 and No.60703046, the Program for New Century Excellent Talents in University NCET-04-0693, NCET-06-0650 and HUST-SRF No.2007Q021B, and the US NSF under Grant No.CCF-0621493, CCF-0621526, CCF-0754951, CNS-0723093, DRL-0737583.

References

- [1] Hewlett-packard labs. <http://tesla.hpl.hp.com/>.
- [2] Lawrence livermore national labs. <http://www.llnl.gov/>.
- [3] Umass trace repository. <http://traces.cs.umass.edu/>.
- [4] B. Anderson. Mass storage system performance prediction using a trace-driven simulator. In *Proceedings of the 22nd IEEE Conference on Mass Storage Systems and Technologies (MSST)*, 2005.
- [5] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5:835–846, 1997.
- [6] A. Dainotti, A. Pescape, and G. Ventre. Worm traffic analysis and characterization. In *Proceedings of the IEEE International Conference on Communications (ICC'07)*.
- [7] M. W. Garrett and W. Willinger. Analysis, modeling and generation of self-similar vbr video traffic. In *Proceedings of SIGCOMM'94*.
- [8] M. Gomez and V. Santonja. Analysis of self-similarity in i/o workload using structural modeling. In *Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, College Park, Maryland, 1999.
- [9] M. Gomez and V. Santonja. A new approach in the modeling and generation of synthetic disk workload. In *Proceedings of the 9th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2000.
- [10] S. Gribble, G. Manku, and E. Brewer. Self-similarity in high-level file systems: Measurement and applications. In *Proceedings of the ACM SIGMETRICS'98*.
- [11] B. Hong and T. Madhyastha. The relevance of long-range dependence in disk traffic and implications for trace synthesis. In *Proceedings of the IEEE Conference on Mass Storage Systems and Technologies*, 2005.
- [12] Z. Kurmas, K. Keeton, and K. Mackenzie. Synthesizing representative i/o workloads using iterative distillation. In *Proceedings of the 11th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2003.
- [13] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the self-similar nature of ethernet traffic (extended version. *IEEE/ACM Transactions on Networking*, 2:1–15, Feb. 1994.
- [14] Z. J. Liu and et al. *Computational Science Technique and Matlab*. Science Press, Beijing, P. R. China, 2001.
- [15] J. P. Nolan. Parameterizations and modes of stable distributions. *Stat. and Prob. Letters*, 38:187–195, 1998.
- [16] Norros. On the use of fractional brownian motion in the theory of connectionless networks. *IEEE JSAC*, 15:200–208, 1997.
- [17] J. Oly and D. Reed. Markov model prediction of i/o request for scientific application. In *Proceedings of the 2002 International Conference on Supercomputing*.
- [18] V. Paxson and S. Floyd. Wide-area traffic: The failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, June 1995.
- [19] A. S. M. Priscilla and et al. A traffic characterization procedure for multimedia applications in converged networks. In *Proceedings of the 13th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'05)*.
- [20] J. Rosinski. On the structure of stationary stable processes. *The Annals of Probability*, 23:1163–1187, 1995.
- [21] G. Samorodnitsky and M. Taqqu. *Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance*. Chapman and Hall, New York, 1994.
- [22] C. Stathis and B. Maglaris. Modelling the self-similar behaviour of network traffic. *Computer Networks*, 34:37–47, 2000.
- [23] D. Surgailis, J. Rosinski, and et al. Stable generalized moving averages. *Probability Theory and Related Fields*, pages 543–558, 1993.
- [24] L. Tian, D. Feng, H. Jiang, and et al. Pro: A popularity-based multi-threaded reconstruction optimization for raid-structured storage systems. In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST)*, San Jose, CA, 2007.
- [25] M. Wang, A. Ailamaki, and C. Faloutsos. Capturing the spatio-temporal behavior of real traffic data. In *IFIP International Symposium on Computer Performance Modeling, Measurement, and Evaluation*, 2002.
- [26] M. Wang, T. Madhyastha, and et al. Data mining meets performance evaluation: Fast algorithms for modeling bursty traffic. In *Proceedings of the 16th International Conference on Data Engineering (ICDE)*, 2002.
- [27] J. Zhang, A. Sivasubramaniam, H. Franke, N. Gautam, Y. Zhang, and S. Nagar. Synthesizing representative i/o workloads for tpc-h. In *Proceedings of HPCA*, 2004.