TECHNICAL REPORT

CSD-TR25-2000

The structural cause of file size distributions

Allen B. Downey

Department of Computer Science

Wellesley College
Wellesley, Massachusetts 02181-8289

The structural cause of file size distributions

Allen B. Downey

October 2000

Abstract

We propose a user model that explains the shape of the distribution of file sizes in local file systems and in the World Wide Web. We examine evidence from 562 file systems, 38 web clients and 4 web servers, and find that this model is an accurate description of these systems, and a better description than an alternative that has been proposed, the Pareto model. We conclude that the distribution of file sizes is generally roughly lognormal and therefore not long-tailed. We discuss the implications of this conclusion for proposed explanations of self-similarity in the Internet.

1 Introduction

Numerous studies have reported traffic patterns in the Internet that show characteristics of self-similarity (see [16] for a survey). These observations have led to a call for an explanatory model of self-similar network traffic [14].

The best models to date are based on the assumption that the distribution of transfer times in the network is long-tailed [19] [17] [22] [9]. In turn, this assumption is based on the assumption that the distribution of file sizes is long-tailed [15] [7].

We contend that the distribution of file sizes in most systems fits the lognormal distribution. We support this claim with empirical evidence from a variety of systems and also with a model of user behavior that explains why file systems tend to have this structure.

We argue that the proposed model is a better fit for the data than the long-tailed model, and furthermore that our user model is more realistic than the explanations for the alternative. We conclude that the distribution of file sizes is not long-tailed.

This result creates a problem for existing explanations of self-similarity in the Internet. We discuss the implications and suggest alternate approaches.

1.1 What does "long-tailed" mean?

"Long-tailed" is defined differently in different contexts. For example, Sigman adopts a definition in which the lognormal distribution is considered long-tailed

[20]. In the context of self-similarity, a long-tailed distribution must have a hyperbolic tail; that is

$$P(X > x) \sim cx^{-\alpha} \quad \text{as} \quad x \to \infty \tag{1}$$

where X is a random variable, c is a normalizing parameter, and α is a shape parameter that determines how long-tailed the distribution is.

This definition is appropriate because this is the asymptotic behavior that is required to produce self-similarity (iff $\alpha < 2$) [21]. By this definition, the lognormal distribution is not long-tailed [19].

2 Distribution of file sizes

In this section we propose a model of the operations that create new files and show that a simulation of this model yields a distribution of file sizes that is a good match for the distributions that appear on real file systems.

Then we show that the simulator is equivalent to a numerical method for solving a partial differential equation (PDE). We show that the solution of this PDE is the analytic form of the result of the simulation. Finally, we use this analytic form to estimate the parameters of observed distributions and measure the goodness-of-fit of the model.

We find that real file systems fit the model well, some surprisingly well. We conclude that the user behavior described by our model explains the observed shape of the distribution of file sizes.

2.1 User model

Thinking about how users behave, we can list the most common operations that create new files:

copying: The vast majority of files in most file systems were created by copying, either by installing software (operating system and applications) or by downloading from the World Wide Web.

translating and filtering: Many new files are created by translating a file from one format to another, compiling, or by filtering an existing file.

editing: Using a text editor or word processor, users add or remove material from existing files, sometimes replacing the original file and sometimes creating a series of versions.

Thus we assert that many file-creating operations can be characterized as linear file transformations: a process reads a file as input and generates a new file as output, where the size of the new file depends on the size of the original.

This assertion suggests a model for the evolution of a file system over time: assume that the system starts with a single file with size s^*, and that users repeat the following steps

1. Select a file size, s, at random from the current distribution of file sizes.

2. Choose a multiplicative factor, f, from some other distribution.

3. Create a new file with size fs and add it to the system.

It is not obvious what the distribution of f should be, but we can make some assumptions. First, we expect that the most common operation is copying, so the mode of f should be 1. Second, thinking about filtering and translations, we expect that it should be about as common to double the size of a file or halve it; in other words, we expect the distribution of f to be symmetric on a log axis.

In Section 2.4 we show that, due to the Central Limit Theorem, the shape of this distribution has little effect on the shape of the resulting distribution of sizes. For now we will choose a distribution of f that is normal on a log scale with mean equal to 1 and an unspecified variance.

There are, then, two parameters in this model, the size of the original file, s^*, and the standard deviation of the distribution of f, which we call γ. By experimenting with a range of parameter values, we observed that s^* acts as a location parameter—it determines the mode of the final distribution—and γ acts as a shape parameter—it determines the dispersion of the distribution.

Next we will see if this model describes real systems. In November 1993 Gordon Irlam conducted a survey of file systems. He posted a message on Usenet asking UNIX system administrators to run a script on their machines and mail in the results. The script uses the `find` utility to traverse the file system and report a histogram of the sizes of the files. The results are available on the web [11].

In June 2000, we ran this script on one of our workstations, a Pentium running Red Hat Linux 6.0. There were 89937 files on the system, including the author's home directory and web pages, but also including the operating system and applications.

Figure 1 shows the cumulative distribution function (cdf) of the sizes of these files, plotted on a log x-axis. The figure does not show the 877 files with length 0, because the logarithm of 0 is undefined.

The figure also shows the cdf of file sizes generated by the simulation, using parameters that were tuned by hand to yield the best visual fit. Clearly the model is a good match for the data, suggesting that this model is descriptive of real systems.

2.2 The analytic model

There are two problems with this model so far: first, it provides no insight into the functional form of these distributions, if there is one; second, it does not provide a way to estimate the model parameters based on real data.

A solution to both problems comes from the observation that the simulator is effectively computing a numerical solution to a partial differential equation

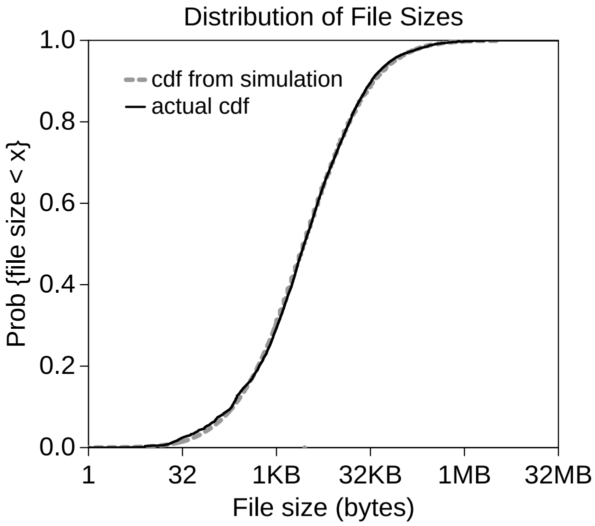

Figure 1: cdf of file sizes on the author's workstation. The dashed line is the cdf of file sizes generated by a simulation of the file creation process.

(PDE). If we can identify the PDE that is being solved and then solve it analytically, we hope that the form of the solution might be the functional form of the distribution of file sizes.

The PDE is the diffusion, or heat equation: $u_t = k^2 u_{xx}$, where $u(x,t)$ is the probability density function (pdf) of file sizes as a function of x, which is the logarithm of the file size, and t, which represents time since the file system was created. k is a constant that controls the rate of diffusion.

Since x represents the logarithm of a nonzero integer value, the range is from 0 to ∞ (we will ignore zero length files). The initial condition in time is the delta function with a peak at the initial file size, $\zeta = log(s^*)$.

The boundary condition at $x = 0$ is not obvious, since the model does not include a meaningful description of the behavior of small files. Fortunately, this boundary behavior is usually irrelevant, so we can choose arbitrarily between $u(0,t) = 0$, which means that files smaller than 1 byte would simply not occur, and and $u_x(0,t) = 0$ which would cause these small files to "reflect" back into the range. In either case the equation can be solved analytically using a Fourier transform [3], and in either case the solution is approximately

$$u(x,t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\zeta}{\sigma}\right)^2\right] \tag{2}$$

where $\sigma = \sqrt{2kt}$. In other words, the distribution under a logarithm transform is Gaussian with mean ζ and standard deviation that increases with k, the rate of diffusion, and t, time. Our model of user behavior provides no way to estimate k, or even to map t onto real time, so there is no real-world meaning for σ. It

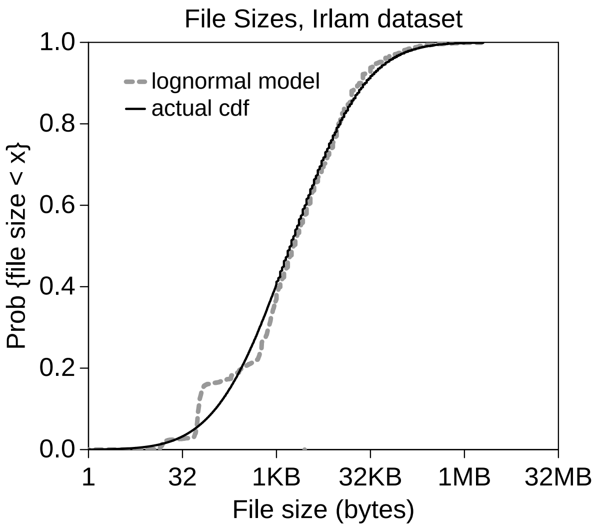

Figure 2: cdf of file sizes on a machine that participated in the Irlam survey. The dashed line is the estimated model.

is a free parameter.

This observation leads us to an easy way to estimate ζ and σ. Given a list of files sizes s_i, we can use the mean of $x_i = log(s_i)$ as an estimate of ζ, and the standard deviation of x_i as an estimate of σ. In the next section we will use this technique to fit analytic models to a collection of datasets.

2.3 Is this model accurate?

Of course, it is possible that the author's file system is unusual, or that the proposed model of user behavior is really a model of the author's behavior. Fortunately, Irlam's survey provides data from 656 machines in a variety of locations and environments. Of these, we discarded 43 because they contained no files with non-zero length, and an additional 52 because they contained fewer than 100 files.

For the remaining 561 file systems, we estimated parameters and then compared the analytic distributions with the empirical distributions. As a goodness-of-fit metric we used the Kolmogorov-Smirnov statistic (KS), which is the largest vertical distance between the fitted and actual curves.

In the best case KS is 1.4 percentiles. For comparison, the fitted model in Figure 1 has a KS of 2.7; there are 8 other cases that are as good or better. The median value of KS is 8.7, indicating that the typical system fits the model well. In the worst case KS is 40, which indicates that the model is not a good description of all systems. The systems that deviate most from the model tend to have few files. Among the 277 systems with more than 5000 files, the worst KS value is 24.

The dataset with the median value of KS is shown in Figure 2. The deviation from the model in this case is typical of what goes wrong in other cases; there is a second mode between 32 and 64 bytes. In general, though, the agreement is good between the tail of the cdf and the model.

Irlam's survey also includes data from one DOS machine. We fitted a lognormal model to this data and calculated a KS statistic of 5, which indicates that this model fits at least one non-UNIX file system.

We conclude that this model is a good description of many real systems, with the qualification that for some purposes it might be more accurate to extend the model by including a mixture of lognormal modes.

2.4 Is this model realistic?

The assumptions the model makes about user behavior are

1. New files are always created by processing an existing file in some way, for example by copying, translating or filtering.
2. The size of the derivative file depends on the size of the original file. We call the ratio of the final size to the original size f.
3. The distribution of f is approximately lognormal with mean 0.

The first assumption is not literally true; there are many other ways files might be created. For example, a new file might be the concatenation of two or more existing files. Or the size of a new file might not depend on an existing file at all.

The second assumption is based on the intuition that many file operations are linear; that is, they traverse the input file once and generate output that is proportional to the size of the input. Again, this is not always true.

The third assumption is based on the intuition that it is as easy, and as common, to double the size of a file as it is to halve it. It is not obvious that this is true, but it is empirically testable.

To find the distribution of multipliers in a real system, we can take advantage of a feature of the `emacs` text editor: when it updates a file, it creates a backup file with the same name as the original, postpended with a tilde. By searching for files ending with a tilde, we can locate a sample of files that have been edited and compare their sizes before and after.

On the author's file system, there are 1025 files with names ending with a tilde; for 989 of these the corresponding original file is available. For these files we computed the ratio of the current size to the size of the backup.

Figure 3 shows the cdf of these ratios on a log scale. In 65 cases the original and the updated files are the same size, so there is a mode at 1. As expected, the distribution is roughly symmetric in log space, although there is a small skew toward larger values (it is more common for files to grow than shrink). The dashed line in the figure is a lognormal curve fitted by matching the moments

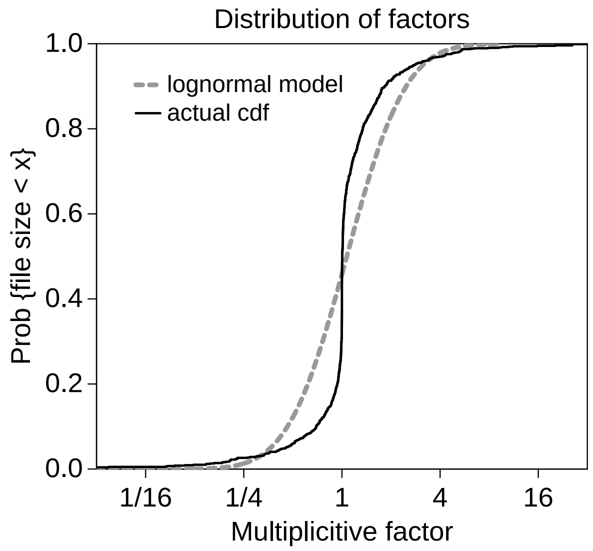

Figure 3: cdf of the ratios of updated file size and original file size for 989 files on the author's system. The dashed line shows a normal curve with the same mean and variance.

of the sample in log space. It shows that the actual curve is significantly more leptokurtotic than a Gaussian.

It turns out, though, that this deviation has almost no effect on the shape of the size distribution. Figure 4 shows a simulation starting with $\zeta = 11$ and using the empirical cdf of factors from Figure 3. The black curves show the cdf of file sizes after 10, 100, and 100000 files were created. The dashed gray line shows a lognormal model fitted to the final curve.

This result suggests that the lognormal model is robust to variation in the distribution of factors. The underlying reason is the Central Limit Theorem. The size of the nth file added to the system depends on the size of one of its predecessors, and the size of the predecessor depends on the size of one of *its* predecessors, and so on. Since the predecessors are chosen at random, the number of predecessors for the nth file grows as $log_2 n$. Thus, the size of the nth file is

$$s_n = s^* \cdot f_1 \cdot f_2 \cdot ... \cdot f_m \tag{3}$$

where m is the number of predecessors for the nth file and the f_i are the random multiplicative factors. Taking the logarithm of this equation yields

$$log(s_n) = log(s^*) + log(f_1) + log(f_2) + ... + log(f_m) \tag{4}$$

In log space, the distance of the nth file from the mean is the sum of m random variables, each of which is roughly normally-distributed. The Central Limit Theorem says that as m goes to infinity, the distribution of this sum converges to normal, provided that the logarithms of the f_i are independent and have

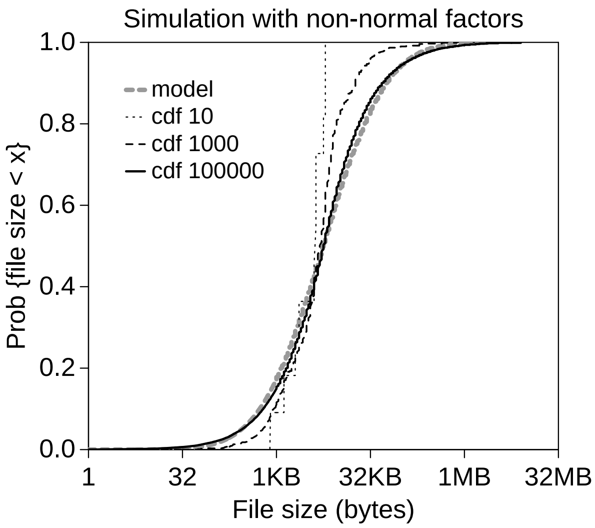

Figure 4: A simulation of file system evolution using the empirical cdf of multiplicative factors.

finite moments. Therefore the distribution of the sum (Equation 4) is normal, and the distribution of the product (Equation 3) is lognormal.

Because m grows slowly (logarithmically) as n increases, convergence may be slow if the distribution of factors is far from lognormal. Furthermore, as the number of files increases, it takes longer and longer for the shape of the distribution to change. For both these reasons, the simulated distribution of files sizes does not quite converge to lognormal, even after 100000 files.

3 The Pareto model of file sizes

Several prior studies have looked at the distribution of file sizes, including local file systems [11] and the World Wide Web [8] [1]. The consensus of these reports is that the tail of these distributions is well-described by the Pareto distribution, and therefore that the distribution of file sizes is long-tailed.

In this section we review these studies and compare the Pareto model to the lognormal model. We find that the lognormal model is a better description of these datasets than the Pareto model, and conclude that there is no evidence to support the claim that the distribution of file sizes is long-tailed.

3.1 Empirical evidence for long tails

In a strict sense, we can never say that an observed distribution is long-tailed, since the definition of long-tailed is a statement about the behavior of the distribution as x goes to infinity, which we can never observe.

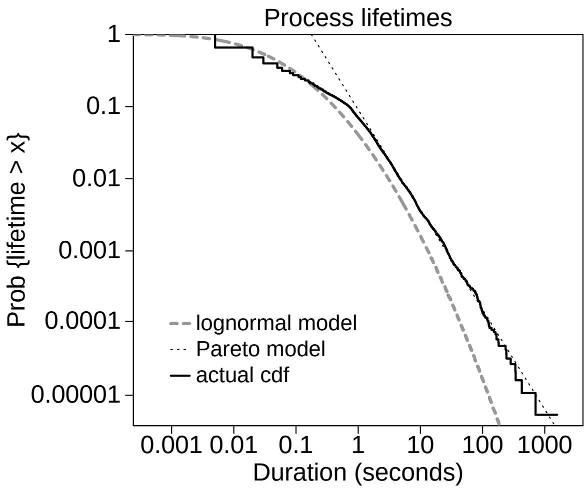

Figure 5: ccdf of process lifetimes from a UNIX workstation.

On the other hand, if there is an underlying process that yields a long-tailed distribution, then there are empirical behaviors we expect to see in a finite sample. For example, if the underlying distribution is Pareto, and we plot the complementary cumulative distribution function (ccdf) on a log-log scale, we expect to see a straight line, at least in the tail behavior, and at least out to the boundary of the measurement. If a sample exhibits this behavior, we have an empirical reason believe that the underlying distribution is long-tailed.

For example, Harchol-Balter and Downey reported that the distribution of lifetimes for processes on UNIX workstations is long-tailed [10]. They collected the lifetimes of 184612 UNIX processes; Figure 5 shows their ccdf, along with a Pareto model chosen by least squares fit to the tail and a lognormal model chosen by the estimation method in Section 2.2. The agreement between the Pareto model and the data suggests that the distribution of lifetimes is long-tailed.

We have to use caution in interpreting this sort of test, though. If we accept the conclusion that a distribution is Pareto, then it is with the qualification that the actual distribution may deviate from the Pareto model beyond the boundary of the measurement. Similarly, if we observe a deviation from the model we should consider whether it is due to a real deviation in the system or a limitation of the measurement. Finally, if the deviation is real, we should consider whether it matters. For example, if the deviation occurs with very low probability, then the system might run for a long time before the deviation could have any effect.

In the following sections we examine file size distributions and look for evidence that they are or are not long-tailed.

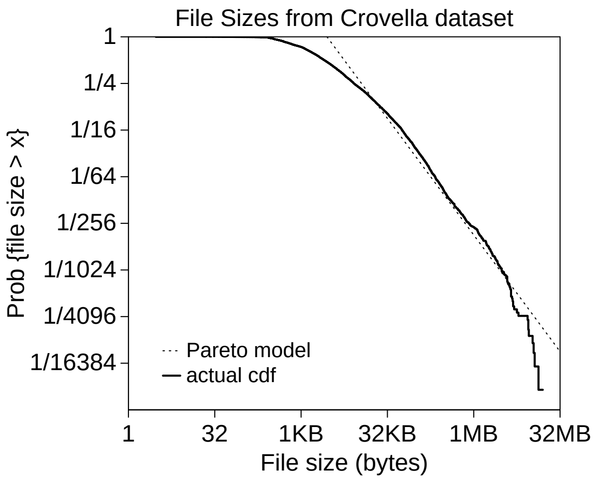

Figure 6: ccdf of unique file sizes from a trace of web file accesses, plotted on a log-log scale. The dotted line shows a Pareto model intended to fit the tail.

3.2 File sizes on the web, client's view

Crovella and Bestavros suggested that the sizes of files available on the World Wide Web have a heavy-tailed distribution with a tail that is approximately Pareto [5]. Based on that observation, Carlson and Doyle proposed a physical model in which web designers, trying to minimize download times, might divide the available information into files such that the distribution of file sizes would obey a power law [4].

In this section we review their data and conclude that the lognormal distribution is a better description than the Pareto distribution, and also that the diffusion model is a better explanatory model than Carlson and Doyle's optimization model.

In 1995 Crovella et al. instrumented web browsers in computer labs at Boston University to collect traces of the files accessed and their access times [8] [5] [7]. From these traces they extracted the list of unique file names and their sizes, and argued that the distribution of these files approximates the distribution of files available on the Web. To test whether the distribution is long-tailed, they plotted its ccdf and applied a graphical test based on the Central Limit Theorem, described below.

We obtained these traces from their web pages and performed the same analysis, yielding 36208 unique files. Figure 6 shows the ccdf of their sizes. This figure is visually similar to the one that appears in [5].

The dotted line in the figure shows a Pareto model that fits the distribution between 4KB and 4MB, which includes about 25% of the files. Based on this fit, Crovella et al. argue that this distribution is long-tailed.

At the same time, they acknowledge disturbing features in this graph: specifically, the apparent curvature of the cdf and its divergence from the model for

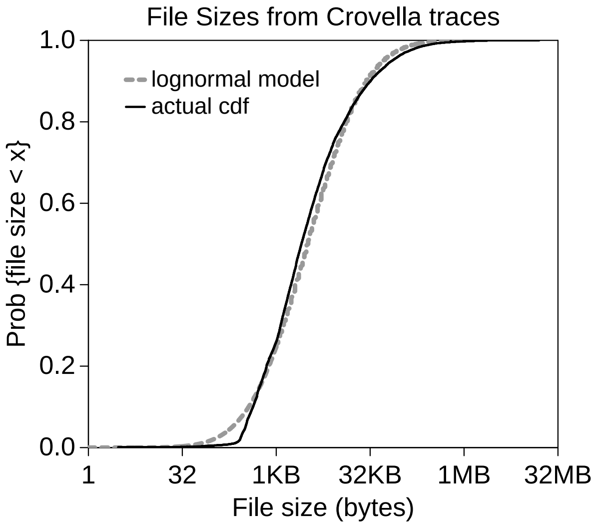

Figure 7: cdf of unique file sizes, plotted on a log axis. The dashed line shows a two-parameter lognormal model.

files larger than 4 MB. Since the divergence is based on a small fraction of the data (only 24 files) it is hard to say whether it is significant. To address this problem, Crovella and Bestavros proposed a test for infinite variance based on the Central Limit Theorem (CLT). Again, the CLT shows that the sum of n random variables chosen independently from a distribution with finite mean and variance will be approximately normally distributed, converging to normal as $n \to \infty$. Conversely, if the variance of the distribution is infinite, then the variance of the aggregate distribution will also be infinite (and therefore not normal).

Using the same dataset of file sizes, they plot aggregate ccdfs for $n = 10, 100, 500$ and show that the aggregate distributions display behavior that is more similar to the behavior of a Pareto distribution than a lognormal distribution. Based on this evidence, they conclude that the "assumption of infinite variance seems justified for this dataset."

We address this conclusion by developing a lognormal model for the data and reevaluating the CLT test.

Figure 7 shows the cdf of file sizes plotted on a log x axis. The dashed line shows a two-parameter lognormal model fitted using the method from Section 2.2. It is not clear that this model fits any better than the Pareto model; the biggest discrepancy is that there are too few files smaller than 256 bytes.

This observation suggests that there is something about files on the web that imposes a minimum file size. We speculate that it is HTML, which requires a variety of formatting statements for even the simplest files. For example, here is the HTML for a common error message:

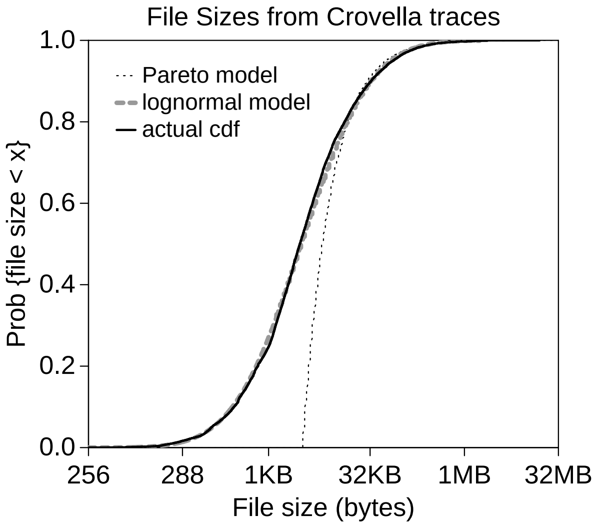

Figure 8: cdf of unique file sizes, shifted by 256 bytes and plotted on a log x-axis. The dashed line shows a three-parameter lognormal model. The dotted line shows a Pareto model.

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>404 Not Found</TITLE>
</HEAD><BODY>
<H1>Not Found</H1>
The requested URL /test was not found on this server.<P>
</BODY></HTML>
```

The page contains only a single line of text, but it requires 198 bytes.

The diffusion model can be extended to handle distributions with an arbitrary lower bound, using the three-parameter form of the lognormal distribution. The third parameter is a shift, θ, applied before the log transform, so that $log(x_i - \theta)$ is distributed normally, where the x_i are samples from a random variable.

We estimated θ visually (256 bytes) and then estimated the other parameters using the two-parameter lognormal model. There are more rigorous ways to estimate the three-parameter model [12], but they are not necessary for our purposes.

Figure 8 shows the transformed cdf along with the model estimated to fit it. The cdf and the model are all but identical. The figure also shows the Pareto model, which bears little resemblance to the measured cdf.

The quality of this fit does not, by itself, demonstrate that the underlying distribution is lognormal (or even that there is an underlying distribution). Similarly, the discrepancy with the Pareto model is not necessarily fatal, especially since the Pareto model was only intended to describe the tail.

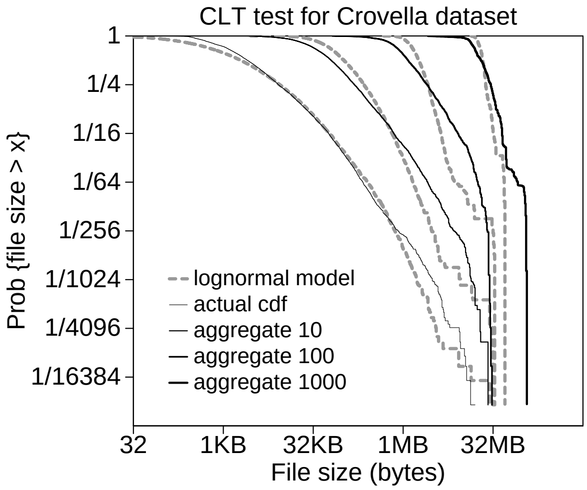

Figure 9: Aggregate ccdfs compared to a lognormal model.

We now address the CLT test proposed by Crovella and Bestavros. To perform this test, we chose a two-parameter lognormal model and a Pareto model that fit the tail well. Next we generated 135357 random values from each of the fitted distributions, and examined their behavior as we aggregated the data with $n = 10, 100, 1000$.

Figure 9 compares the aggregate behavior of the lognormal model with the actual sample of file sizes. In general, the lognormal model captures the slope of the actual curves, although there are discrepancies in the location. Figure 10 shows the actual data compared to the Pareto model. At each level of aggregation there are substantial discrepancies between the model and the actual behavior. We are inclined to say that this test supports the lognormal model, but it is not conclusive.

It may seem odd that we have applied the same test Crovella and Bestavros used and reached the opposite conclusion. There are two reasons. The first is that they examined a lognormal model with parameters $\zeta = 2, \sigma = 2$ and characterized its aggregate behavior as the general behavior of lognormals. The lognormal model we examined ($\zeta = 10, \sigma = 3.4$) has a much higher variance, which means that it takes longer to converge to a normal distribution as n increases, which makes it more difficult to distinguish from a long-tailed distribution. Also, they aggregated data in non-overlapping chunks, which means that the number of samples at aggregation level n was $135357/n$. We aggregated data in overlapping chunks so the number of samples at aggregation level n was $135357 - n$. As a result, our aggregated cdfs show more of the tail behavior, where the discrepancies with the Pareto model are most clear.

We are including the CLT test in order to explain the difference between our conclusions and theirs, but we do not believe that the CLT test has empirical power to distinguish between distributions with finite and infinite variance. The

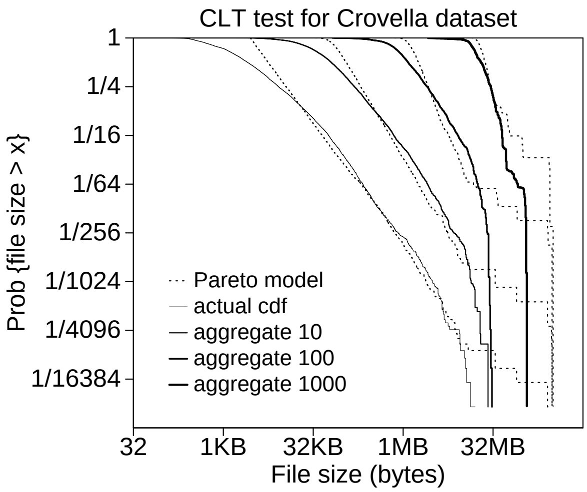

Figure 10: Aggregate ccdfs compared to a Pareto model.

deviations at higher aggregation levels are just numerical amplifications of the deviation that was already noted between the actual cdf and the model; they do not provide additional evidence in favor of either model. This conclusion is consistent with [6] which shows that the scaling behavior of sums from a lognormal distribution is visually indistinguishable from that of a long-tailed distribution.

We conclude that the traces collected by Crovella et al. provide little empirical support for the Pareto model of file sizes and strong support for the lognormal model.

Furthermore, we believe that the diffusion model is more realistic than Carlson and Doyle's explanatory model for the Pareto distribution. Their model is based on the assumption that the material available on web pages is "a single contiguous object," that the website designers are free to divide into files, and that they do so such that the the files with the highest hit rates are the smallest.

Carlson and Doyle acknowledge that "this view of website design is so idealized that it may not explain in any detail why real web sites have power law distributions."

3.3 File sizes on the web, server's view

Between October 1994 and August 1995, Arlitt and Williamson [1] collected traces from web servers at the University of Waterloo, the University of Calgary, the University of Saskatchewan, NASA's Kennedy Space Center, ClarkNet (an ISP) and the National Center for Supercomputing Applications (NCSA).

For each server they identified the set of unique file names and examined the distribution of their sizes. They report that these data sets confirm the observation by Crovella et al. that the distributions match the Pareto model,

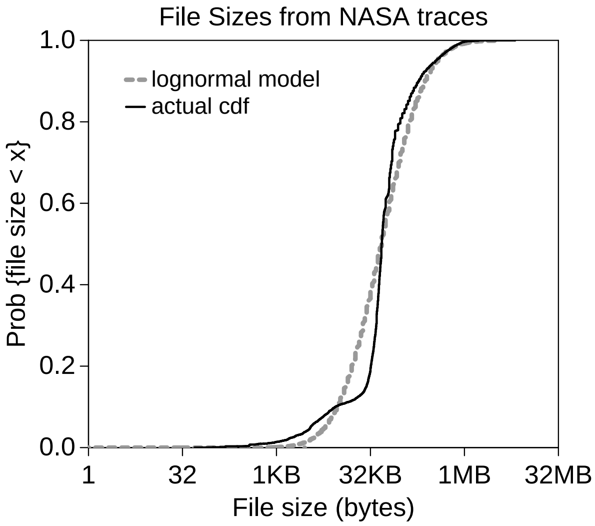

Figure 11: cdf of file sizes from the NASA dataset. The dashed line shows a lognormal model.

and they give Pareto parameters for each dataset. They do not, however, provide evidence that these models fit the data.

Four of these traces are in the Internet Traffic Archive (`http://ita.ee.lbl.gov`). We processed these traces by extracting each HTTP GET command that was successful (return code 200) and that returned more than zero bytes of data. For each of these we recorded the name and size of the file.

To derive the set of distinct files, we weren't sure whether to include files with the same name but different sizes. We decided that these cases probably represent successive versions of the same file, so we decided to include all versions as distinct files. Using either definition of "distinct" we get size cdfs that are visually similar to those in [1], although the number of distinct files is significantly different from the numbers reported there.

We fitted a two-parameter lognormal model to each dataset and calculated the KS statistics. For the Calgary, ClarkNet and Saskatchewan datasets the model fits well (KS values 5.8, 5.5, and 9.7); for the NASA dataset it doesn't (KS value 19). We present the NASA dataset as a worst-case example.

Figure 11 shows the cdf of file sizes from the NASA dataset and the fitted lognormal model. As we have seen before, there are significant deviations from the model; in particular there is an additional mode centered at 4 KB. Nevertheless, the model is a good description of the general shape of the cdf.

To examine the tail behavior, we plotted the ccdf in log-log space (Figure 12). The lognormal model captures the tail behavior well, better than any Pareto model, which would appear as a straight line on these axes.

We conclude that these datasets provide strong support for the lognormal model and no support for the Pareto model. In the next section we discuss the

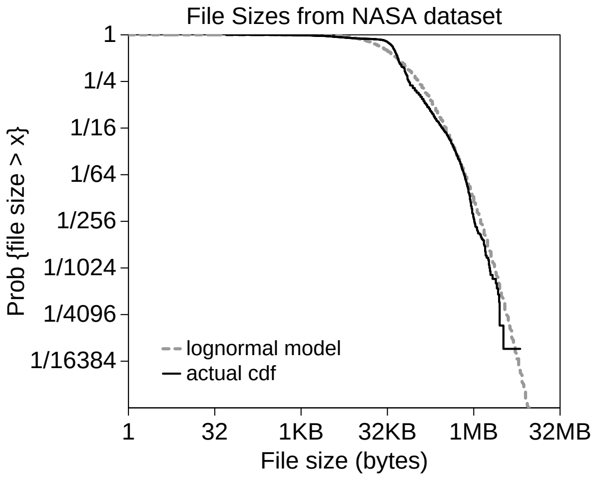

Figure 12: ccdf of file sizes from the NASA dataset, on a log-log scale.

implications of this conclusion for explanatory models of self-similarity in the Internet.

4 Self-similar network traffic

What does it mean to say that network traffic is self-similar? In the strictest sense, this is not an empirical claim, since the definition of self-similarity is a statement about behavior as time scales go to infinity. But if we have reason to believe that there is an underlying process that yields self-similar Internet traffic, then there are a number of behaviors we expect to see:

- Visual self-similarity. If self-similar traffic is plotted on a range of time scales, it appears visually similar regardless of scale.
- Statistical self-similarity. Some summary statistics of self-similar traffic are invariant over all time scales.
- Long-range dependence. Self-similar processes are one kind of long-range dependent process; therefore we expect self-similar traffic to exhibit significant autocorrelations even with large lags.

Researchers have proposed a variety of techniques that test for evidence of self-similarity, including variance-time plots, R/S statistics and periodograms [13] [2], Whittle's procedure and Beran's goodness-of-fit [19], and local and global scaling analysis [9].

As evidence of long-range dependence and self-similarity has accumulated, researchers have been looking for an explanatory model.

One of the first models was an M/G/∞ queueing model in which network transfers are customers with Poisson arrivals and the network is an infinite-server system [19] [17]. In this model, if the distribution of service times is Pareto then the number of customers in the system is an asymptotically self-similar process.

Willinger et al. proposed an alternative that models users as ON/OFF sources in which ON periods correspond to network transmissions and OFF periods correspond to inactivity [22]. If the distribution of the lengths of these periods is long tailed, then as the number of sources goes to infinity, the superposition of sources yields an aggregate traffic process that is fractional Gaussian noise, which is self-similar at all time scales.

In the original model the lengths of the ON and OFF periods were identically-distributed, long-tailed random variables. A companion paper proves that the model yields self-similarity if either the distribution of ON periods *or* the distribution of OFF periods is long-tailed [21].

Two subsequent papers have extended this model to include a realistic network topology, bounded network capacity and feedback due to the congestion control mechanisms of TCP [15] [9]. In both cases, the more realistic models yielded qualitatively similar results.

All of these models are based on the assumption that the distribution of transfer times in the Internet is long tailed, but there is currently no independent evidence that it is. In fact, Paxson reported that the distribution of data bytes transferred during ftp sessions fits a lognormal model [18].

Crovella et al. have suggested that the distribution of transfer times depends at least in part on the distribution of available file sizes, and that the cause of long-tailed transfer times is long-tailed file sizes [7]. However, we have argued that the distribution of file sizes is not long-tailed. Therefore, it cannot be the structural cause of self-similarity in the Internet.

In order for these models to explain self-similarity, they have to explain the origin of long-tailed transfer times. There are several possibilities:

- Even if file sizes are not long tailed, transfer times might be. The performance of wide-area networks is highly variable in time; it is possible that this variability causes long-tailed transfer times.
- Even if the length of individual sessions is not long-tailed, the lengths of bursts might be. From the network's point of view there is little difference between a TCP timeout during a transfer and the beginning of a new transfer [19].
- The distribution of interarrival times might be long-tailed.

A final possibility is that network traffic is not truly self-similar. Paxson and Floyd show that in the $M/G/\infty$ model, if the distribution of service times is lognormal, the resulting count process is not self-similar and not long-range dependent. On the other hand, over a wide range of time scales—possibly all time scales that matter—such a process may be statistically indistinguishable from a truly self-similar process.

As ongoing work we are investigating these possibilities to try to find the ultimate cause of self-similarity in Internet traffic.

Acknowledgements

Many thanks to Mark Crovella (Boston University) and Carey Williamson (University of Saskatchewan) for making their datasets available on the web, Gordon Irlam for his survey of file sizes, and to Peter Danzig (University of Southern California), Jeff Mogul (Digital's Western Research Lab), Vern Paxson (Lawrence Berkeley National Lab), and Mike Schwartz (University of Colorado at Boulder) for organizing the Internet Traffic Archive. Thanks also to Kim Claffy and Andre Broido (CAIDA), Rich Wolski (University of Tennessee) and Lewis Rothberg (University of Rochester) for reading drafts of this paper and making invaluable comments.

References

[1] Martin F. Arlitt and Carey L. Williamson. Web server workload characterization: the search for invariants. In *ACM SIGMETRICS'96*, pages 126–137, May 1996.

[2] Jan Berand, Robert Sherman, Murad Taqqu, and Walter Willinger. Long-range dependence in variable-bit rate video traffic. *IEEE Transactions on Communications*, 43(2/3/4):1566–1579, 1995.

[3] Paul W. Berg and James L. McGregor. *Elementary Partial Differential Equations*. Holden-Day, San Francisco, 1966.

[4] Jean M. Carlson and John Doyle. Highly optimized tolerance: a mechanism for power laws in designed systems. *Physics Review E*, 60(2), August 1999.

[5] Mark E. Crovella and Azer Bestavros. Self-similarity in world-wide web traffic: evidence and possible causes. In *ACM SIGMETRICS'96*, pages 160–169, May 1996.

[6] Mark E. Crovella and Murad S. Taqqu. Estimating the heavy tail index from scaling properties. *Methodology and computing in applied probability*, 1(1):55–79, 1999.

[7] Mark E. Crovella, Murad S. Taqqu, and Azer Bestavros. Heavy-tailed probability distributions in the World Wide Web. In *A Practical Guide To Heavy Tails*, pages 3–26. Chapman & Hall, 1998.

[8] Carlos R. Cunha, Azer Bestavros, and Mark E. Crovella. Characteristics of WWW client-based traces. Technical Report BU-CS-95-010, Boston University, 1995.

[9] Anja Feldman, Anna C. Gilbert, Polly Huang, and Walter Willinger. Dynamics of IP traffic: a study of the role of variability and the impact of control. In *ACM SIGCOMM'99*, pages 301–313, 1999.

[10] Mor Harchol-Balter and Allen B. Downey. Exploiting process lifetime distributions for dynamic load balancing. In *ACM SIGMETRICS'96*, pages 13–24, May 1996.

[11] Gordon Irlam. Unix file size survey, `http://www.base.com/gordoni/ufs93.html`. Accessed 31 May 2000, 1994.

[12] Norman Johnson, Samuel Kotz, and N. Balakrishnan. *Continuous univariate distributions, Vol. 1 & 2.* John Wiley and Sons, New York, 1994.

[13] Will E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, February 1994.

[14] Kihong Park. Future directions and open problems in performance evaluation and control of self-similar network traffic. In *Self-Similar Network Traffic and Performance Evaluation*, chapter 21. Wiley-Interscience, 2000. Preprint.

[15] Kihong Park, Gitae Kim, and Mark E. Crovella. On the relationship between file sizes, transport protocols, and self-similar network traffic. Technical Report BU-CS-96-016, Boston University, August 1996. Short version appeared in Proc. Fourth Int'l Conference on Network Protocols.

[16] Kihong Park and Walter Willinger. Self-similar network traffic: An overview. In *Self-Similar Network Traffic and Performance Evaluation*, chapter 1. Wiley-Interscience, 2000. Preprint.

[17] M. Parulekar and A. M. Makowski. $M/G/\infty$ input process: a versatile class of models for network traffic. Technical Report T.R. 96-59, Institute for Systems Research, 1996.

[18] Vern Paxson. Empirically-derived analytic models of wide-area TCP connections. *IEEE/ACM Transactions on Networking*, 2(4):316–336, August 1994.

[19] Vern Paxson and Sally Floyd. Wide-area traffic: the failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3:226–244, 1995.

[20] Karl Sigman. A primer on heavy-tailed distributions. *Queueing Systems*, 33(1–3):261–273, 1999.

[21] Murad S. Taqqu, Walter Willinger, and Robert Sherman. Proof of a fundamental result in self-similar traffic modeling. *Computer Communication Review*, 27:5–23, 1997.

[22] Walter Willinger, Murad S. Taqqu, Robert Sherman, and Daniel V. Wilson. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. In *ACM SIGCOMM'95*, pages 100–113, 1995.