# Metrics for Mass-Count Disparity

Dror G. Feitelson
School of Computer Science and Engineering
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

## Abstract

*Mass-count disparity is the technical underpinning of the "mice and elephants" phenomenon — that most samples are small, but a few are huge — which may be the most important attribute of heavy-tailed distributions. We propose to visualize this phenomenon by plotting the conventional distribution and the mass distribution together in the same plot. This then leads to a natural quantification of the effect based on the distance between the two distributions. Such a quantification addresses this important phenomenon directly, taking the full distribution into account, rather than focusing on the mathematical properties of the tail of the distribution. In particular, it shows that the Pareto distribution with tail index $1 < a < 2$ actually has a relatively low mass-count disparity; the effects often observed are the result of combining some other distribution with a Pareto tail.*

## 1. Introduction

Heavy tails have been found to characterize the distributions of many aspects of computer-related systems. This means that while most elements are very small, some are very big and may even dominate the observations of the system. In many cases, this disparity among numerous small items and few large items is the main factor that affects system performance, and is more important than the exact mathematical description of the shape of the distribution's tail. It is therefore beneficial to have a set of simple and intuitive metrics that describe and quantify this effect.

Examples of system designs where heavy tails play an important role include the following. The heavy-tailed distribution of file sizes justifies the Unix inode structure, with its direct support for small files and indirect links that are only used for very large files [4, 13]. The heavy-tailed distribution of process runtimes can be exploited for load balancing in a cluster, by focusing on migrating the longest processes [15, 11]. The heavy-tailed distribution of Internet flows has led to the proposal that only the large flows need to be monitored for accounting purposes [8]. Heavy tails also affect caching strategies: caching to minimize the number of requests forwarded to a server should focus on requests for small pages, whereas caching to minimize the number of bytes requested from the server should focus on the large ones [3].

The apparent importance of heavy tails has prompted considerable debate about the best way to model them [7, 19, 6]. However, the practical optimizations for heavy-tailed workloads cited above do not depend on the precise characteristics of the tail of the distribution. Rather, they rely on the distinction between the body of the distribution and its tail, and more specifically on the phenomenon of mass-count disparity: that a small number of items account for the majority of mass, whereas all small items together only account for negligible mass [5, 10] (or, using concrete examples from computer workloads, a typical process is short, but a typical second of CPU activity is part of a long process; a typical file is small, but a typical byte of storage belongs to a large file). This disparity is sometimes referred to as the "mice and elephants" phenomenon. But this metaphor may conjure the image of a bimodal distribution[1], which could be misleading: in most cases, the progression is continuous.

Our contribution in this paper is to suggest a way to visualize mass-count disparity by a combined plot of the count and mass distributions. Based on this, we propose metrics that quantify the effect. One is a generalization of the proverbial 20/80 rule. Another two are based on what we call the 0/50 rule. Furthermore, our suggested metrics are non-parametric, so they do not depend on the underlying distribution, and can be easily calculated for an empirical distribution. Thus they sidestep the whole issue of what distribution provides the best fit to the data. Using these metrics we can make several interesting observations regarding heavy-tailed workload distributions.

[1] A typical mouse weighs about 28 grams, whereas an elephant weighs 3 to 6 tons, depending on whether it is Indian or African. Cats, dogs, and zebras, which fall in between, are missing from this picture.

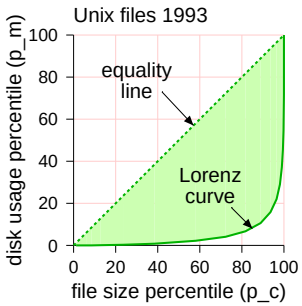


**Figure 1.** *A Lorenz curve for the Unix 1993 data. The Gini coefficient corresponds to the shaded area.*

## 2. Mass-Count Disparity Plots

A typical way to look at a distribution is by plotting its CDF (cumulative distribution function) $F_c(x) = \Pr(X < x)$. We call this the "count" distribution, because we are counting how many items, say files, there are of each size (hence the subscript $c$). Another way to look at the same data is to weight each file by its size. In effect this creates a distribution of bytes rather than files: instead of specifying the probability that a file be smaller than $x$, it specifies the probability that a byte *belong to* a file smaller than $x$. We call this the "mass" distribution, because here we are looking at the bytes that make up the files. In mathematical notation, assuming a pdf $f(x)$, the mass distribution is calculated as [5]

$$F_m(x) = \frac{\int_{-\infty}^{x} x' f(x')\, dx'}{\int_{-\infty}^{\infty} x' f(x')\, dx'}$$

Mass-count disparity occurs when these two distributions are very different from each other. This need not be the case. If mass is concentrated in one main mode, the two distributions will be similar. For example, this happens when data comes from a normal distribution (Fig. 8 below). But if the tail of the distribution contains a disproportional fraction of the mass, while the body contains a disproportional fraction of the items, the count and mass distributions will diverge.

One way to compare the two distributions is to create a derived graph showing the relationship between their percentiles — essentially a P-P plot [14]. For each value of $x$, we find the percentile of $x$ in the two distributions: $p_c = F_c(x)$ and $p_m = F_m(x)$. We can then plot $p_m$ as a function of $p_c$:

$$p_m = F_m(\, F_c^{-1}(p_c)\,)$$

This is called the Lorenz curve, and is used in economics to measure equality (or rather, inequality) in the distribution of wealth [17]. If wealth is equally distributed, the $p$ percentile of the population also controls $p$ percent of the wealth, so $p_m = p_c$ — a straight diagonal line. But if wealth is unequally distributed, and the poorest $p$ percentiles of the population control less than $p$ percent of the wealth, we will find that $p_m < p_c$. A measure of inequality is therefore the degree to which the curve diverges from the diagonal, which can be measured by the area between the curve the the diagonal, a metric called the Gini coefficient[2]. Of course, the same procedure can be applied to other types of data. Fig. 1 shows the Lorenz curve corresponding to the file sizes found in a large-scale survey of Unix systems from 1993 [13]. Such a graph was used by Crovella to show that the few largest files may account for some 60% of the total bytes in a file system [5]; similarly, Arlitt and Williamson showed that 10% of documents were responsible for 80–95% of requests from different web servers [3].

However, graphs like the Lorenz curve are somewhat hard to interpret. Here we suggest a simpler approach: simply plot the two CDFs together. Doing this for the Unix files data leads to the plot shown in Fig. 2. The difference between the distributions is immediately apparent: in the count (files) distribution, most of the weight is at low values around a kilobyte, whereas in the mass (disk space) distribution the weight is spread out at higher values around several megabytes. Plotting the two distributions together has been done before, e.g. in [16, 18]. However, they did not use this to derive simple metrics as we do next.

## 3. Metrics for Mass-Count Disparity

Mass-count disparity is often described in words by quoting certain percentiles of the distributions. For example, Harchol-Balter et al. describe a heavy-tailed distribution of files requested from a Web server as "the largest $<$3% or the requested files make up $>$50% of the total load... 50% of the files have size less than 1K bytes. 90% of files have size less than 9.3 K bytes" [12].

As an alternative, we suggest the use of mass-count disparity plots as an inspiration for two simple "rules". One is the proverbial 20/80 or 10/90 rule (also known as the "Pareto principle"). Its application to the Unix files data is shown on the top of Fig. 2. The graph shows that the data is very close to the 10/90 rule: 10% of the files are big files, and account for 90% of the disk space, and vice versa. The boundary between big and small in this case is 16 KB. Less extreme data sets would be closer to a ratio of 20/80.

An even more dramatic demonstration of mass-count disparity is the 0/50 rule. As shown on the bottom of Fig. 2, a full *half* of the files are so small that together they account for a negligible fraction of the disk space. At the same time, half of the disk space is occupied by a very small fraction

[2] Actually the Gini coefficient is defined to be twice this area, so as to normalize it to the range [0..1].

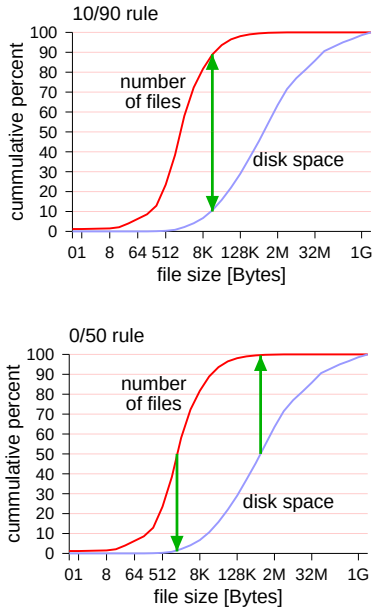


**Figure 2.** *The distribution of Unix file sizes from 1993, showing the 10/90 and 0/50 rules.*

of files, which are each very large. This is the property that is often the most important in the context of computer systems, as it allows one to focus on that part of the workload that is responsible for most of the mass, and ignore the numerous items that do not account for much [11, 5, 10, 8].

Formalizing this idea, we can suggest several numerical measures that indicate the degree to which a distribution indeed displays mass-count disparity. While our names for these metrics are new, the metrics themselves have been used by others, e.g. by Irlam in his descriptions of the Unix files data from 1993 [13].

The simplest metric is the *joint ratio*. This is a direct generalization of the 10/90 rule and the 20/80 rule. The 10/90 rule, for example, says two things at once: That 10% of the items account for a full 90% of the mass, and also that 90% of the items account for only 10% of the mass. The generalization is to find the percentage $p$ such that $p$% of the items account for $100-p$% of the mass, and $100-p$% of the items account for $p$% of the mass. The smaller $p$ is, the greater the mass-count disparity. To find the joint ratio, note that CDFs are non decreasing, so the complement of a CDF is non-increasing. Therefore there is a unique $x$ that satisfies the condition

$$F_c(x) = 1 - F_m(x)$$

Given this $x$, compute $R = 100F_m(x)$. The joint ratio is then $R/(100-R)$.

The 0/50 rule is generalized by two metrics. In practice, the 0 here is not really 0; the metrics quantify how close to 0 we get. The first metric is $N_{1/2}$, which quantifies the percentage of items from the tail needed to account for half of the mass:

$$N_{1/2} = 100(1 - F_c(\, F_m^{-1}(0.5)\,))$$

The second is $W_{1/2}$, and quantifies the total mass of the bottom half of the items:

$$W_{1/2} = 100F_m(\, F_c^{-1}(0.5)\,)$$

Note that all these three metrics measure the vertical distance between the two distributions. This is because the vertical distance best characterizes the mass-count disparity. But it may also be interesting to know how much larger the tail items are. This can be measured by the *median-median distance*, that is the distance between the medians of the two distributions. The farther apart they are, the heavier the tail of the distribution. As absolute values depend on the units used, it makes sense to express this distance as a ratio (or take the log of the ratio in order to express the distance as the number of orders of magnitude that are spanned).

## 4. Real-World Examples

The above metrics are readily applied to myriad examples of data that displays mass-count disparity. We start with two types of highly skewed distributions that characterize computer workloads: distributions of size and distributions of popularity.

Size-related examples are shown in Fig. 3. On the top is data about files from Unix file systems: the survey conducted in October 1993 by Irlam, representing about 12 million files from over a thousand file systems, and a local file system (Hebrew University computer science) with over 18 million files sampled in June 2005. These two datasets are amazingly similar both in terms of their general shapes and in terms of the specific values assumed by our metrics. Thus it is possible that this data is indeed representative and stable. However, it is desirable to collect data from additional contemporary installations in order to verify this.

On the bottom is data on Unix process runtimes: a dataset used in simulations by Harchol-Balter and Downey [11], with about 185,000 processes that arrived over 8 hours (from 9AM to 5PM) to a server at CMU in November 1994, and a dataset of nearly 450,000 Unix processes from a departmental server at Hebrew University computer science, covering about a month during October-November 2005[3].

These two datasets are quite different from each other. One noticeable difference is that in the 1994 data 34% of the processes were tabulated as requiring 0 time; in the

[3]One process was removed from this dataset, as it was tabulated to have run for 341 hours, which seems extremely unlikely; the next highest runtime is 29 hours.

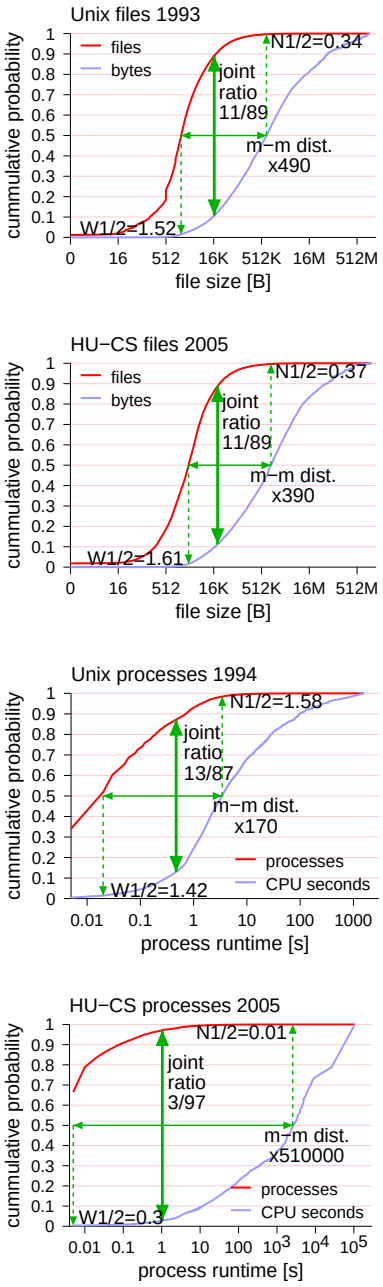


**Figure 3.** *Examples of the joint ratio, $N_{1/2}$, $W_{1/2}$, and median-median distance metrics for size data.*

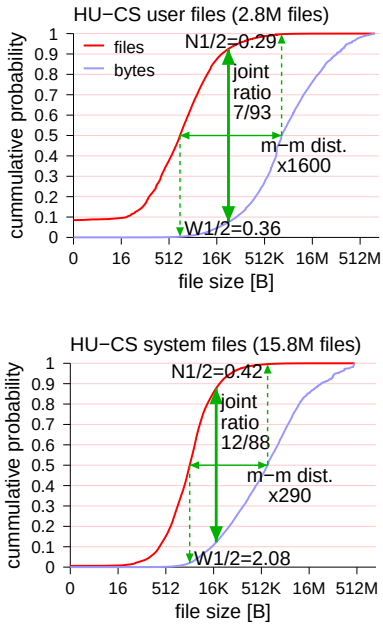


**Figure 4.** *File size data for user home directories vs. system files, from the Hebrew University data set.*

2005 data this doubled to 66%. The reason for this is probably the limited resolution of 0.01 seconds used by the last-comm command that was used to collect the data (and in fact, this is the best possible with a 100Hz operating system clock). As processor speeds increase, this resolution becomes less and less adequate for tabulating short processes. As a side-effect, this limitation also taints two of the 2005 metrics. The median-median distance is measured relative to the 0.005 seconds used to represent all the processes that are tabulated as taking 0 time; this is not necessarily the true value for the (unknown) real median. And the $W_{1/2}$ metric should be interpreted as saying that half the processes use *at most* 0.3% of the CPU time, because this fraction is actually attributed to 66.5% of the processes, not to only half of them.

Regardless of these qualifications, the newer dataset is much more extreme than the previous one. However, it is risky to jump to conclusions based on this change, because both data sets are from a single installation. We can only say that it is necessary to collect additional data from multiple installations in order to learn more about the distribution of process runtimes.

Interesting observations can be made by applying the metrics to subsets of the data, rather than looking at the whole dataset monolithically. For example, the HU-CS file sizes data shown above is actually the composition of two distinct data sets: user files and system files. Fig. 4 shows the distributions and metrics for each group independently. This shows that the distribution of user files is much more extreme (in terms of its mass-count disparity) than the distribution of system files; for example, the joint ratio for user files is 7/93, whereas for system files it is only 12/88. Together with the $N_{1/2}$ and $W_{1/2}$ metrics this enables us to quantify and describe the difference between the distributions of the two subsets. The metric values shown above for the complete dataset are essentially an average of the two subsets.

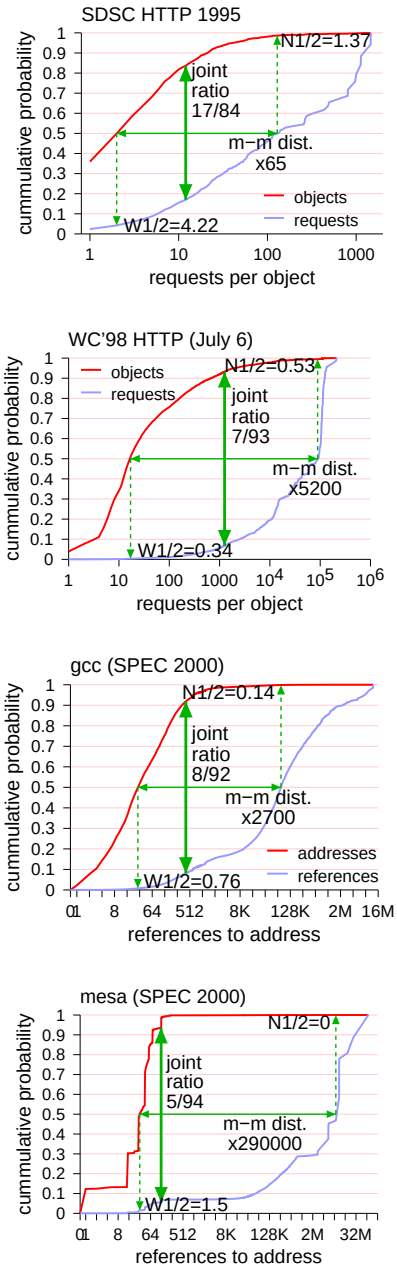


**Figure 5.** *Examples of the joint ratio, $N_{1/2}$, $W_{1/2}$, and median-median distance metrics for popularity data.*

Examples of datasets relating to popularity are shown in Fig. 5. The top graphs show the popularity of different files in HTTP logs: the SDSC HTTP log collected by Polterock et al. in August 1995, with some 28 thousand requests from one day, and data from the 1998 World Cup website collected by Arlitt [2]; the full dataset contains over 1.3 billion requests spanning 3 months; here we show 16.7 million requests from one day.

Popularity in web traces is measured by the number of requests for the same file[4]. The count distributions show how many files were requested $x$ times, while the mass distributions show how many requests were directed at these files. Obviously, these data display a strong mass-count disparity. The World Cup data is much more extreme than the SDSC data, indicating that the SDSC requests were more evenly spread, whereas the traffic to the world cup site was much more focused on a select set of documents. This distinction is probably more related to the very different types of installations and not to the progress in time from 1995 to 1998.

The bottom two plots in Fig. 5 show data regarding locality of reference from two SPEC benchmarks: gcc and mesa. Memory address traces from the execution of these and other applications indicate that some data structures and variables are much more popular than others; this is part of the effect of temporal locality. Our metrics indicate that this indeed leads to very strong mass-count disparity. But even more striking is the wide diversity of results. With data from only 7 different benchmarks, the observed joint ratios ranged from 1/99 to 8/92, and the observed median-median distances ranged from a factor of 2,700 to a factor of 1,200,000. Moreover, the distributions tended to display various special features that are not captured by the metrics, such as the modal structure apparent in the mesa benchmark.

It is expected that analogous plots can be drawn for measurements of additional quantities, such as function-level application profiles: most functions are called a small number of times, and most calls are to a small subset of the functions. However, at present we do not have such data available.

As with the size data, interesting observations can be made by applying the metrics to subsets of the data. For example, the world cup 1998 log actually contains data about requests fielded over nearly 3 months. The top plot in Fig. 6 shows the popularity data of this whole data set. The second plot shows the data limited to the initial portion of the log: more than 70 million requests, all during the month of May, well before the games started. The bottom two plots show data for June 30, one of the most intensive days in the middle of the tournament. This shows that the results for the complete dataset are an average between the results for the pre-games period, which is less extreme, and the peak period, which is much more extreme. Interestingly, the metric that best brings out the difference is the median-median distance, which grows from a factor of about 1700 before or between games to about 17,000 during peak activity. This indicates that during the games the requests were much more focused on a relatively small set of pages

[4]We include both successful requests (status 2XX) and consistency requests (status 304, meaning no change).

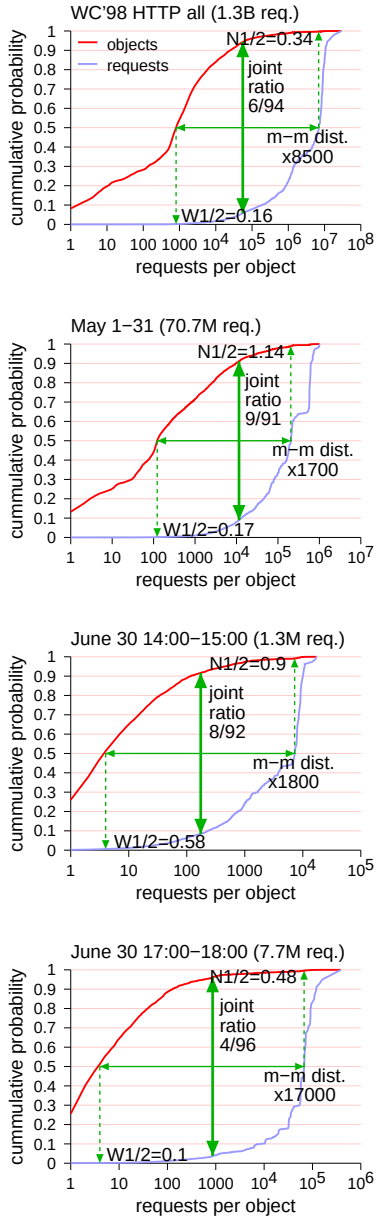


**Figure 6.** *Popularity of files on the WC'98 server, for different durations.*

— probably the ones relating to the games being played at that time. Note this difference is even observed on a short timescale of a few hours, as witnessed by the bottom two plots. Thus it can serve to characterize the well-known phenomenon of "flash crowds" [1].

Naturally, mass-count disparity plots can be drawn for other types of data that are unrelated to computer workloads. For example, Fig. 7 shows data regarding downloads of open-source projects from the SourceForge repository[5],

[5]We only used projects that had at least one download, and ignored the 80597 that had 0 downloads (about $\frac{2}{3}$ of the listed projects). The data is from May 2005.

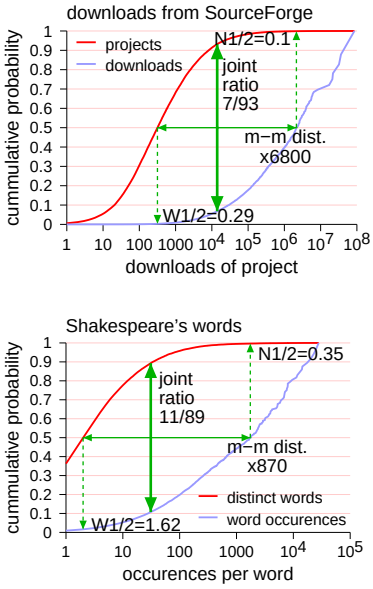


**Figure 7.** *Mass-count disparity in the popularity of open-source projects on SourceForge, and words in Shakespeare's plays.*

and word-frequency data from Shakespeare's plays. It is well-known that the word frequency distribution is highly skewed: most words appear very few times in any given corpus, but some words (e.g. "the", "and", "in") appear lots of times [21, 20]. And indeed, this leads to significant mass-count disparity. Other obvious candidates for such plots are the distribution of wealth, which was the topic studied by Pareto, Lorenz, and Gini, or the distribution of city sizes.

## 5. Mass-Count Disparity of Distributions

The above examples show the mass-count plots for empirical data, and extract the metrics from these plots. But given a mathematical expression for a distribution, it may also be possible to compute the metrics analytically.

For example, consider the exponential distribution. The count distribution is simply the well-known CDF, $F_c(x) = 1 - e^{-x/\theta}$. The mass distribution can be easily calculated as

$$F_m(x) = \frac{\int_0^x \frac{x'}{\theta} e^{-x'/\theta} dx'}{\int_0^\infty \frac{x'}{\theta} e^{-x'/\theta} dx'} = 1 - \frac{x+\theta}{\theta} e^{-x/\theta}$$

Given these equations, we can calculate the different metrics. For example, the joint ratio occurs at the $x$ value for

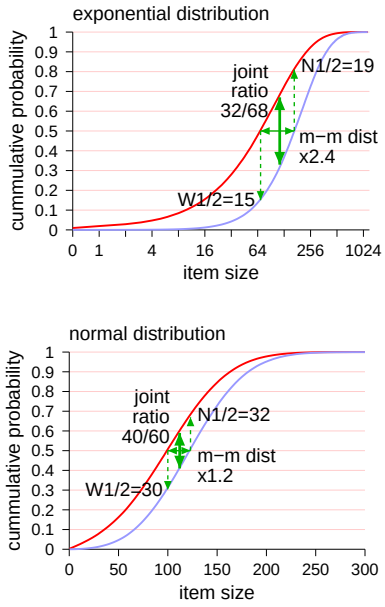


**Figure 8.** *Mass-count plots and metrics for the exponential and normal distributions, which are not heavy-tailed. The exponential distribution has mean $\theta = 100$. The normal distribution has mean $\mu = 100$ and standard deviation $\sigma = 50$, so it is a bell shape situated just to the right of $x = 0$. Note the use of a linear scale for this plot.*

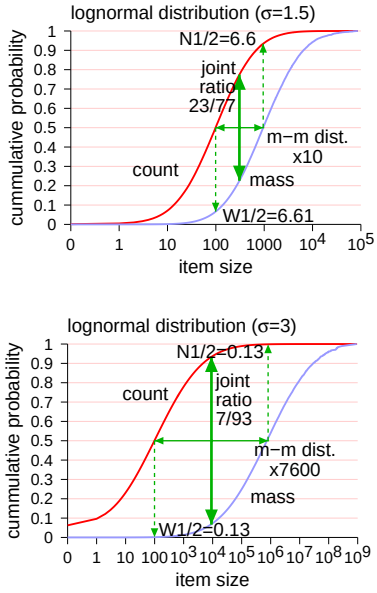


**Figure 9.** *Mass-count plots and metrics for the long-tailed lognormal distribution.*

which the sum of the two distributions is 1. For $\theta = 1$ this happens at $x = 1.15$. Using this value, we find that $F_c(1.15) = 0.683$. The joint ratio for the exponential distribution is therefore 32/68, indicating a rather low mass-count disparity. An even lower metric value is obtained for the normal distribution, as illustrated in Fig. 8.

Naturally, mass-count disparity does exist in distributions that have a heavier tail. For example, Fig. 9 shows data for the lognormal distribution, which has a long tail, meaning that it decays subexponentially [9]. Both the shown distributions have $\mu = 4.6 = \ln(100)$, and indeed for both the median of the count distribution is near 100. The difference is in the dispersion, as expressed by the $\sigma$ parameter. The bigger the dispersion, the heavier the tail, and the stronger the mass-count disparity.

With the Pareto distribution, the mass-count disparity depends on the tail index. The Pareto count distribution is

$$F_c(x) = 1 - \left(\frac{k}{x}\right)^a$$

where $k$ is the minimal value possible (that is, the distribution is defined for $x \geq k$), and $a$ is the tail index. The mass distribution is

$$F_m(x) = \frac{\int_0^x x' \frac{ak^a}{x'^{a+1}}\,dx'}{\int_0^\infty \frac{ak^a}{x'^{a+1}}\,dx'} = 1 - \left(\frac{k}{x}\right)^{a-1}$$

The integrals only converge for the case when $a > 1$; if $a$ is smaller, the tail is so heavy that the mean is undefined. This is reflected in the shape of the plots (Fig. 10). When $a$ is small, a significant part of the mass occurs in the few top samples, and we get very high mass-count disparity. When $a = 1$, the mass distribution is a straight line. When $a$ is larger (the case described by the equations) the metric values are rather moderate. In particular, the values for $a = 1.5$ are rather similar to those of the exponential distribution shown in Fig. 8. However, the shapes of the two distributions are quite different, and of course their tails are also quite different.

The reason that the Pareto distribution may not exhibit a high mass-count disparity when using our metrics is that the metrics do not focus on the tail, but rather on the relationship between the tail and the body of the distribution. The vast majority of reports regarding heavy tails in the literature are indeed limited to the tails of the observed distributions, and do not claim that the complete distribution is well described by a power law (i.e. that the distribution is Pareto). For example, both the Unix file sizes data from 1993 and the Unix process runtimes data from 1994 have heavy tails, and indeed the right-hand sides of their mass-count disparity plots (Fig. 3) have the concave shape of the plots for the Pareto distribution with $a > 1$. However, they are not well-modeled by a Pareto distribution. The reason

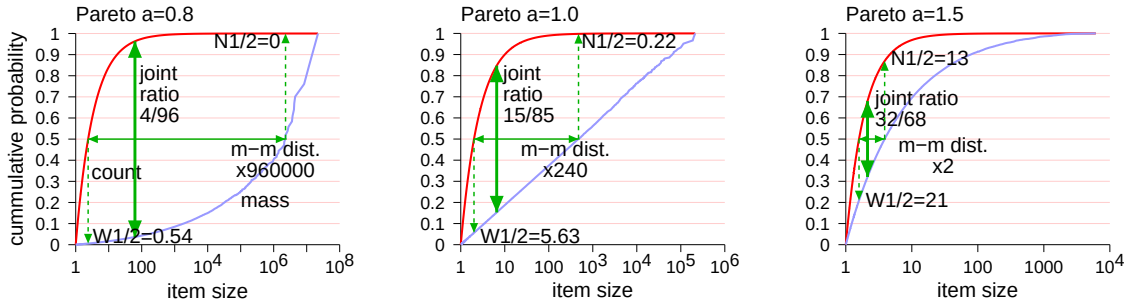


**Figure 10.** *Mass-count plots and metrics for the heavy-tailed Pareto distribution.*

is that these datasets have many more small items than a Pareto distribution does, and these are needed in order to lead to a significant mass-count disparity. A possible way to construct a distribution with such a power-law tail that does exhibit significant mass-count disparity is therefore to combine two distributions — for example, an exponential body and a Pareto tail.

## 6. Limitations

Mass-count disparity plots are very useful for visualizing the effect of mass-count disparity which occurs in many real-life datasets, and our metrics enable this effect to be easily quantified. However, there are some limitations to their applicability.

The main limitation occurs when the distributions in question are modal. For example, if the distribution happens to have a discrete mode exactly at the $x$ value that satisfies the condition for the joint ratio ($F_c(x) = 1 - F_m(x)$), the two CDFs will have a discontinuity and the condition will actually not be satisfied. A possible solution would be to use the values just to the right or just to the left of the mode, but then the two ratios do not coincide.

An example is provided by packet sizes data from a trans-Pacific Internet link. Considering the first 10,000 packets in this trace, nearly two thirds of them had a size of 1500 bytes (which was also the maximal size observed). As a result all our metrics are meaningless. At best, we can replace the joint ratio by saying that the smaller 46% of the packets contain 8.4% of the bytes, while the remaining 64% of the packets account for 91.6% of the bytes.

Note, however, that modal distributions do not always have such a strong detrimental effect. If we consider the first 1,000,000 packets in the above trace, the results are somewhat less extreme: only 40.3% of the packets have a size of 1500 bytes, so the medians do not share a common value, and the $W_{1/2}$ and median-median distance are defined. Still, the joint ratio is not, as the smaller 59.7% of packet contain 15.7% of bytes, while the top 40.3% of packets account for 84.3% of bytes. Interestingly, more recent data (e.g. from 2005) is even less modal, and consequently all of our metrics are well defined.

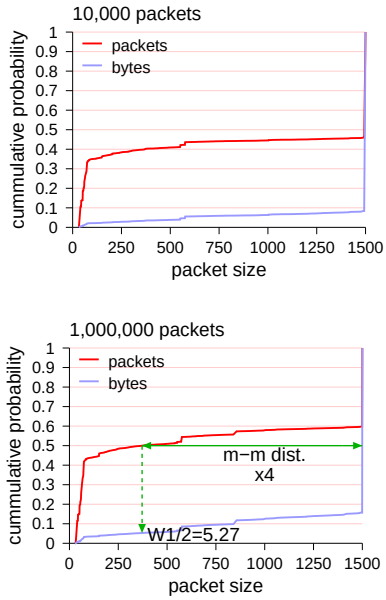


**Figure 11.** *Mass-count plots for highly modal packet size data from samplepoint A of a trans-Pacific T1 line in the WIDE backbone, starting on Friday, September 1, 2000, at 13:59. Note the use of a linear scale.*

Another potential problem occurs when using the median-median distance to characterize short-lived effects, such as flash crowds on the Internet. The problem is that the mass distribution then depends on the window of observation: the longer we observe the effect, the higher the mass associated with the large items from the tail of the distribution. As a result the median-median distance becomes larger as the observation window becomes longer. But if it becomes too long, and extends beyond the duration of the effect, the count distribution starts to catch up and the

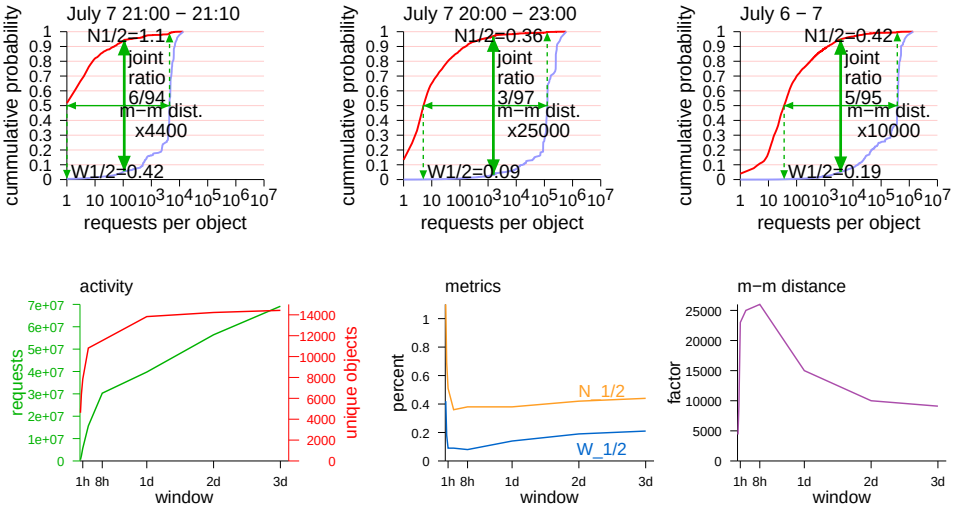


**Figure 12.** *Mass-count plots and metrics for the HTTP WC'98 data, with different ranges around the "flash" of activity related to the semi-finals in the evening of July 7, 1998.*

median-median distance is reduced again.

The above is demonstrated in Fig. 12, using data from the 1998 world cup web site. The first semi-final game occurred in the evening hours of July 7, 1998, and there was no other peak of activity in the two days that preceded it. As the figure shows, the joint ratio is very stable at between 3/97 and 6/94 for all observation windows. The $N_{1/2}$ and $W_{1/2}$ metrics are also rather stable provided the observation window is not too short (e.g. 10 minutes). But the median-median distance is very sensitive to the observation window. Note, however, that this sensitivity can also be exploited to identify such effects.

Finally, another potential limitation is that mass-count disparity plots require data about the complete distribution. In some cases it is much easier to obtain data about the unique large-scale items from the tail of the distribution than about the abundant items from the body of the distribution. In particular, it may be difficult to assess how many small items there are, which is crucial for a correct rendition of the count distribution. For example, when studying data about flows in the Internet, one would need to tabulate all the short flows at wire speed, which could be very difficult [8]. Another (non-computer related) example is the distribution of wealth, where data is typically tabulated using a rather coarse classification into income brackets.

## 7. Conclusions

Mass-count disparity is a basic characteristic of skewed distributions, such as many of those that characterize computer workloads. In fact, it may be argued that this characteristic is the most important attribute of such distributions, more so than the exact mathematical properties of the tail of the distribution. This reflects that fact that mass-count disparity characterizes the complete distribution, and in particular, the relationship between the tail and the body of the distribution. In contrast, metrics such as the tail index only characterize the tail, and are oblivious to the relationship between the tail and the body. In particular, we found that the Pareto distribution (with $a > 1$) actually has a relatively low mass-count disparity. To create a distribution with a heavy (power-law) tail and a high mass-count disparity, one needs many more small items.

We have proposed a set of metrics that together quantify salient features of mass-count disparity, based on a comparison of the mass distribution and the count distribution. Applying these metrics to computer workload data shows that significant mass-count disparity indeed exists. But more importantly, these metrics allow for fine distinctions between related workloads. For example, they seem to indicate that workloads are becoming more skewed with time, and that the distributions of popularity on different web servers can be quite different. However, not all metrics have the same power:

- The $N_{1/2}$ and $W_{1/2}$ metrics are the least discriminating. They tend to be extremely small in all the skewed workloads we have examined. Therefore they can serve as an initial metric: if they are more than a few percentage points, the data is not really highly skewed.

- The joint ratio is more discriminatory, and values ranging from close to 20/80 down to 2/98 have been observed. It has the advantage of being quite robust de-

spite changes in the amount of data being considered or its resolution.

- The median-median distance is the most variable, ranging from a factor of a few dozen up to a factor of hundreds of thousands. Such large factors indicate that typical units of mass indeed tend to belong to items that are considerably bigger than the typical items. However, this metric is susceptible to dependence on the size of the dataset being analyzed.

Further research is needed to verify the trends shown here, and to discover additional findings. This first requires the collection of additional datasets from various computer systems. Given enough data from different installations will enable meta-studies that can identify invariants and trends that are generally representative.

## Acknowledgments

This research was supported in part by the Israel Science Foundation (grant no. 167/03).

Thanks to Elliot Jaffe and Danny Braniss for collecting HU file sizes data, and to Yoav Etsion for running simulations of SPEC benchmarks and collecting address popularity data. Thanks also to all those who collected data sets and made them available on the web, especially Gordon Irlam for the Unix 1993 file sizes survey[6], Allen Downey for the 1994 Unix process runtimes data[7], Martin Arlitt for the World Cup 1998 data[8], Joshua Polterock, Hans-Werner Braun, and K Claffy for the SDSC HTTP data[9], The MAWI working group of the WIDE project for the packet-level data of a trans-Pacific T1 line[10], Greg Madey for making the SourceForge data accessible under license from SourceForge[11], and The Internet Shakespeare Editions site for the Shakespeare word frequencies[12].

## References

[1] I. Ari, B. Hong, E. L. Miller, S. A. Brandt, and D. D. E. Long, "*Managing flash crowds on the Internet*". In 11th *Modeling, Anal. & Simulation of Comput. & Telecomm. Syst.*, pp. 246–249, Oct 2003.

[2] M. Arlitt and T. Jin, "*A workload characterization study of the 1998 world cup web site*". *IEEE Network* **14(3)**, pp. 30–37, May/Jun 2000.

[3] M. F. Arlitt and C. L. Williamson, "*Web server workload characterization: the search for invariants*". In *SIGMETRICS Conf. Measurement & Modeling of Comput. Syst.*, pp. 126–137, May 1996.

[4] M. J. Bach, *The Design of the UNIX Operating System*. Prentice-Hall, 1986.

[5] M. E. Crovella, "*Performance evaluation with heavy tailed distributions*". In *Job Scheduling Strategies for Parallel Processing*, D. G. Feitelson and L. Rudolph (eds.), pp. 1–10, Springer Verlag, 2001. Lect. Notes Comput. Sci. vol. 2221.

[6] A. B. Downey, "*Lognormal and Pareto distributions in the Internet*". *Comput. Commun.* **28(7)**, pp. 790–801, May 2005.

[7] A. B. Downey, "*The structural cause of file size distributions*". In 9th *Modeling, Anal. & Simulation of Comput. & Telecomm. Syst.*, Aug 2001.

[8] C. Estan and G. Varghese, "*New directions in traffic measurements and accounting: focusing on the elephants, ignoring the mice*". *ACM Trans. Comput. Syst.* **21(3)**, pp. 270–313, Aug 2003.

[9] C. M. Goldie and C. Klüppelberg, "*Subexponential distributions*". In *A Practical Guide to Heavy Tails*, R. J. Adler, R. E. Feldman, and M. S. Taqqu (eds.), pp. 436–459, Birkhäuser, 1998.

[10] M. Harchol-Balter, "*Task assignment with unknown duration*". *J. ACM* **49(2)**, pp. 260–288, Mar 2002.

[11] M. Harchol-Balter and A. B. Downey, "*Exploiting process lifetime distributions for dynamic load balancing*". *ACM Trans. Comput. Syst.* **15(3)**, pp. 253–285, Aug 1997.

[12] M. Harchol-Balter, B. Schroeder, N. Bansal, and M. Agrawal, "*Size-based scheduling to improve web performance*". *ACM Trans. Comput. Syst.* **21(2)**, pp. 207–233, May 2003.

[13] G. Irlam, "*Unix file size survey - 1993*". URL http://www.base.com/gordoni/ufs93.html.

[14] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*. McGraw Hill, 3rd ed., 2000.

[15] W. E. Leland and T. J. Ott, "*Load-balancing heuristics and process behavior*". In *SIGMETRICS Conf. Measurement & Modeling of Comput. Syst.*, pp. 54–69, 1986.

[16] J. L. Lo, L. A. Barroso, S. J. Eggers, K. Gharachorloo, H. M. Levy, and S. S. Parekh, "*An analysis of database workload performance on simultaneous multithreaded processors*". In 25th *Ann. Intl. Symp. Computer Architecture Conf. Proc.*, pp. 39–50, Jun 1998.

[17] M. O. Lorenz, "*Methods of measuring the concentration of wealth*". *Pub. Am. Stat. Assoc.* **9(70)**, pp. 209–219, Jun 1905.

[18] "*MAWI working group traffic archive*". URL http://mawi.wide.ad.jp/mawi/, 2005.

[19] M. Mitzenmacher, "*A brief history of generative models for power law and lognormal distributions*". *Internet Math.* **1(2)**, pp. 226–251, 2003.

[20] H. S. Sichel, "*On a distribution law for word frequencies*". *J. Am. Stat. Assoc.* **70(351)**, pp. 542–547, Sep 1975.

[21] G. K. Zipf, *Human Behavior and the Principle of Least Effort*. Addison-Wesley, 1949.

[6] http://www.base.com/gordoni/ufs93.html
[7] http://www.allendowney.com/research/sigmetrics96/traces.tar.gz
[8] http://ita.ee.lbl.gov/html/contrib/WorldCup.html
[9] http://ita.ee.lbl.gov/html/contrib/SDSC-HTTP.html
[10] http://mawi.wide.ad.jp/mawi/
[11] http://www.nd.edu/~oss/Data/data.html
[12] http://ise.uvic.ca/Annex/Stats/FreqAscending.html