Dependability, Access Diversity, Low Cost: Pick Two

Ming Chen
Microsoft Research Asia

Lex Stein
Microsoft Research Asia

Zheng Zhang
Microsoft Research Asia

Abstract

Storage systems and data centers are growing rapidly and costing more, with higher energy bills. Users want dependable access to a wide variety of diverse content, but providers want to lower costs. Many studies have looked at the tradeoffs between cost and dependability, but few have looked carefully at how content request diversity changes this relationship. In this paper, we model a disk array and develop an analytical framework to study the relationships between dependability, access diversity, and low cost. We show how access diversity changes the relationship between cost and dependability and that all three are in tension with one another. It is possible to improve any two together, but not all three simultaneously.

1. Introduction

Many studies of internet workloads observe poor access locality. Some observe Zipf distributions [1] [5] [3], others worse [2] [4]. Online data sets are growing in size, many faster than even disk capacities, creating larger data centers with larger energy needs.

Users want dependable service, where request times are predictable and perceived performance does not change wildly from one moment to the next. Access diversity as a design objective is not very well-understood. Nor is its relationship to cost and dependability.

This paper constructs an analytical model of a disk array within a general system model. The general model has inputs, outputs, and internal state. The inputs are a parameter knob and request distribution and the outputs are a cost and the response distribution. The objectives of dependability, access diversity, and low cost can all be explained through this general model. Dependability has two properties. First, for given inputs, the response times are predictable and only vary within a narrow range. Second, a small change in the inputs will not induce a wild change in the cost or response distribution. Access diversity is the breadth of accesses across offered content. It measures how widely the request distribution varies across files. Under this definition, diversity is the inverse of locality. In the analysis of this paper, we look exclusively at energy cost because other costs remain equal under the parameters we vary. Cost is an output of the system because it varies under k and the request distribution. This paper explores the relationships among these three objectives.

2. System model

In this section, we present an analytical model of a disk array storage system. The system clusters popular data to reduce power costs. Clustering distributes load to disks in a bimodal way so that disks are either busy or idle all the time. If energy cost is significant, this approach can save a lot of money. Unfortunately, the success of this approach depends on the popularity distribution of files. This section develops a model to study the tradeoffs between dependability, access diversity, and low cost.

The model fixes the number and kind of disks, eliminating capital cost as a varying factor. Energy is the only varying cost. The model does not consider disk failure. In this model, each disk can be independently powered off. Once a disk is off, its data can only be accessed by turning it on then spinning up the platter. Therefore, requests to off disks will incur significantly higher latencies than requests to on disks. The average time to reactivate a disk is denoted by $1/\mu_{spin}$ and these times follow an exponential distribution. The request rate is λ and request intervals follow a Poisson distribution. The workload is read-only. An on disk has a service rate μ_{on} and the service times are exponentially distributed.

Data is allocated to disks using an approach similar to popular data concentration [6]. Disks are partitioned into hot and cold sets. Popular data are allocated to hot disk sets and unpopular data to cold disk sets. We term this *clustering*. Within the hot and cold sets, data are evenly distributed to balance load. To avoid constantly cycling on and off, cold disks wait for a quiescent period before turning off. Note that hot and cold do not mean on and off. Hot disks will always be on, but cold disks may be on or off at any given time. Cold disks try to turn off, while hot disks do not.

The energy consumed with clustering is denoted by e_c.

In addition, for comparison, we evaluate a baseline system. In the baseline, all disks are hot. The energy consumed by the baseline system is denoted by e_b. Let D be the amount of user data and C the storage capacity of a disk. The system reserves some storage space for unexpected use and some service capacity for unexpected loads. Let r_p be the reserved portion of storage space. Let r_v be the reserved portion of service capacity. With these terms and assumptions, the required number of disks is $n = max(D/(c(1-r_p)), \lambda/(\mu_{on} * (1-r_v)))$, where D/C is the number of disks required to store data and $\lambda/(\mu_{on} * (1-r_v))$ is the number of disks required to serve requests. The maximum of these two values determines the required number of disks. Let p be the energy consumption of a disk that is always on. Therefore, the energy consumption of the always-on baseline system is:

$$e_b = pn = p * max(D/(c(1-r_p)), \lambda/(\mu_{on} * (1-r_v)))$$

The symbol m denotes the miss rate of hot disks. Let k be the ratio of hot disks to total disks. All disks hold the same amount of data and all requests are the same size. Therefore, the request rate for a hot disk is $\lambda_h = \lambda(1-m)/(nk)$ and the request rate for a cold disk is $\lambda_c = \lambda m/(n(1-k))$. Frequently cycling disks on and off may impair their reliability, so a disk stays on for time of at least T_o after spinning up. Spinning up incurs extra energy consumption. To compensate for this, the spin-up interval of a cold disk $T_i = 1/\lambda_c$ is twice T_o. Under these conditions and assumptions, the energy consumption with clustering is:

$$\begin{aligned} e_c &= nkp + n(1-k)pT_o/T_i && \text{if } T_i \geq 2T_o \\ e_c &= nkp + n(1-k) = np && \text{if } T_i < 2T_o \end{aligned}$$

Cost and response distribution are the two outputs. To calculate them we use the $M/M/1$ model, which assumes Poisson arrival and exponential service times. This model derives both the mean response time and the standard deviation of the response time. For the baseline, the mean response time is $T = 1/(a-b)$, where a is the service rate and b the request rate. The standard deviation of the response time is $\rho = 1/(a-b)$. With clustering, the mean response time of a hot disk is $1/(\mu_{on} - \lambda_h)$. The mean response time of a cold disk is $1/(\mu_{spin} - \lambda_c)$, if $T_i \geq 2T_o$ and $1/(\mu_{on} - \lambda_c)$, if $T_i < 2T_o$. Of all requests, fraction $1-m$ are served by the hot disks and fraction m are served by the cold disks. Therefore, the mean response time of clustering is:

$$\begin{aligned} T_c &= \frac{1-m}{\mu_{on} - \lambda_h} + \frac{m}{\mu_{spin} - \lambda_c} && \text{if } T_i \geq 2T_o \\ T_c &= \frac{1-m}{\mu_{on} - \lambda_h} + \frac{m}{\mu_{on} - \lambda_c} && \text{if } T_i < 2T_o \end{aligned} \quad (1)$$

denotation	*default value*	*short description*
λ_0	50 requests/s	total system request rate
D	137 TB	total unique data
C	300 GB	disk capacity
μ_{on}	2	service rate of an on-disk
μ_{spin}	1/60	service rate of an off-disk
r_v	0.2	bandwidth reserved for peak
r_p	0.2	storage for unexpected usage

Table 1. analysis: default parameters The values used in analysis. λ_0 of 50 is derived from the Maze trace. There, the average request size is 18MB, for a load of 900MB/s. We assume a sustained disk throughput of 36MB/s, for a μ_{on} of 2 requests per second.

Similarly, we can derive the standard deviation of response time for clustering:

$$\begin{aligned} \rho_c &= \sqrt{(\frac{1-m}{(\mu_{on} - \lambda_h)})^2 + (\frac{m}{(\mu_{spin} - \lambda_c)})^2} && \text{if } T_i \geq 2T_o \\ \rho_c &= \sqrt{(\frac{1-m}{\mu_{on} - \lambda_h})^2 + (\frac{m}{\mu_{on} - \lambda_c})^2} && \text{if } T_i < 2T_o \end{aligned}$$

3. Analysis

The system described in section 2 can be analyzed as a system of the general form described in section 1. The inputs are the workload and request distribution. The parameter values are summarized in table 1. Two are taken from observations of the Maze[1] file-sharing system. The average request size is 18MB, from a trace taken from 2/18/2004 to 3/21/2004. The total unique data is 137TB, from a snapshot taken on 12/27/2006.

Figure 1 shows the system response to inputs. The first column for cost, the second for mean response time, and the third for standard deviation of response time. Each column has three rows, for decreasing request rates (λ) of $\lambda_0/1$, $\lambda_0/10$, and $\lambda_0/100$. In each graph, the outputs are shown by contours on a plane of miss rate versus the ratio of hot to total disks (k).

There are also workload curves, which show how the miss rate (m) of a workload changes with changing k. The curves are Zipf with differing α values. Consider the curve with α of 1.2 in any one of the figures. As k increases, more disks become hot and fewer requests miss the hot disks.

Figures 1(a), 1(d), and 1(g) show how cost varies with the inputs. All three show both decreasing costs in their lower-left regions and a *plain* in their upper regions. The

[1] `http://maze.pku.edu.cn`

pattern of the lower region is explained by the fact that costs drop as either the miss rate or k drops. When the miss rate drops, cold disks receive requests less frequently and are therefore more likely to hibernate longer. A decrease of k means there are fewer hot disks that are always on. Together, these effects reduce energy consumption. In the plains of the upper regions, the cost value is flat at 1, meaning that the system does not save energy over the baseline. This is because higher access diversity spreads load and destroys locality. The cold disks cannot hibernate.

Figures 1(b), 1(e), and 1(h) show how the mean response time output varies with the inputs. All three show both a *mountain* region where mean response time is large and unstable and a large, flat *plain* where mean response time is small and stable. The mountains of 1(e) and 1(h) are caused by the lengthy response time of the off cold disks. When the miss rate drops to these regions, cold disks begin to hibernate because the request frequency is low ($\lambda_0/10$ and $\lambda_0/100$).

In formula 1, if the miss rate, m, is large enough to contribute to the aggregate mean response time, then the mean response can be large. The upper and bottom plains are formed in different ways. In the upper plains, the miss rate is high. The cold disks receive frequent requests and cannot hibernate. This makes the response time low. In the bottom plain, the miss rate is low. Most of the requests are served by hot disks, so the response time is low here too.

Figures 1(c), 1(f), and 1(i) show similar patterns for the standard deviation of response time. They are caused by the same set of factors.

property			
dependability	$+$	$+$	$-$
access diversity	$+$	$-$	$+$
low cost	$-$	$+$	$+$

Table 2. tradeoffs between properties. A '$+$' connotes a desirable change and a '$-$' an undesirable change.

The three objectives are all desirable, but cannot be simultaneously maximized. Figure 1 shows the tension. For dependability, the system should avoid the mountains of the mean and standard deviation graphs. There, mean and standard deviation of response time increase and become more sensitive to the workload and knob inputs. To increase access diversity, the system must move upwards in the m versus k planes. Increasing diversity means decreasing locality. Therefore, for a fixed k, the miss rate will increase. To get low cost, the inputs must move left and downwards in the m versus k planes. In this region, there are fewer hot disks on and the cold disks can hibernate. This is where the system can most reduce energy costs.

The three subfigures with requests rates of $\lambda_0/10$ reveal three tradeoffs. First, for dependability and access diversity, low cost must be sacrificed. Dependability requires workloads to be above or below the mountain, and access diversity moves the workloads upwards. These two forces push the workloads into the top region, making it difficult to get low cost. Second, for dependability and low cost, diversity must be sacrificed. Dependability moves workloads above or below the mountain. Low cost pushes workloads down and to the left. Together, the system is forced downwards, decreasing access diversity. Finally, for access diversity and low cost, dependability must be sacrificed. Diversity moves the inputs upwards and low cost moves them down and to the left. To meet these conflicting requirements, the inputs must enter the mountains, where dependability suffers.

The subfigures with $\lambda_0/1$ and $\lambda_0/100$ show similar behavior. With $\lambda_0/1$, the relationship between access diversity and low cost intensifies. As a result, only less diversity can decrease costs. With $\lambda_0/100$, the relationship between diversity and cost relaxes.

Therefore, we can only pick two of dependability, access diversity, and low cost. Table 2 shows the tradeoffs.

4. Conclusion

This paper develops a general analytical model for a disk array. Within this model, we discovered tradeoffs between the objectives of dependability, access diversity, and low cost. Two can only be simultaneously improved by sacrificing the third. The tension increases under higher load.

References

[1] L. Breslau, P. Cao, L. Fan, G. Philips, and S. Shenker. Web caching and Zipf-like distributions: Evidence and implications. In *Proc. of IEEE INFOCOM 1999*, Mar. 1999.

[2] L. Cherkasova and G. Ciardo. Characterizing locality, evolution, and life span of accesses in enterprise media server workloads. In *Proc. of the 12th NOSSDAV*, May 2002.

[3] S. Gadde, J. Chase, and M. Rabinovich. Web caching and content distribution: A view from the interior. In *Proc. of the 5th International Web Caching and Content Delivery Workshop*, May 2000.

[4] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, Modeling, and Analysis of a Peer-to-peer File-Sharing Workload. In *Proc. of ACM SOSP*, Oct. 2003.

[5] V. N. Padmanabhan and L. Qiu. The content and access dynamics of a busy web site. In *Proc. of ACM SIGCOMM 2000*, Aug. 2000.

[6] E. Pinheiro and P. Bianchini. Energy Conservation Techniques for Disk Array-Based Servers. In *Proc. of ACM/IEEE ICS*, June 2004.

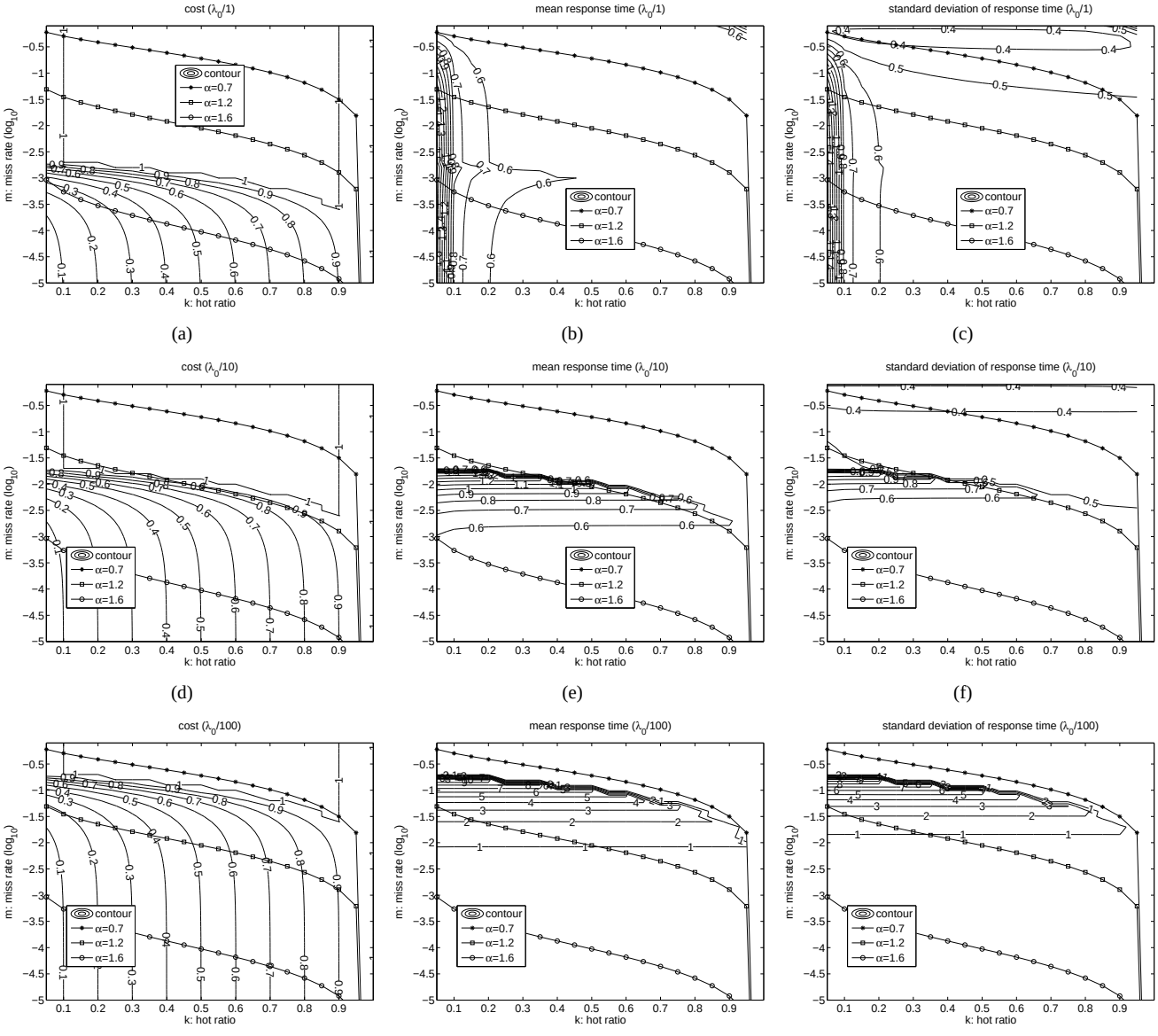

Figure 1. system response to varying inputs (α, λ, k). Cost, mean response time, and standard deviation of response time on the plane of the ratio of hot disks k and miss rate by varying λ. Cost is a fraction of e_c/e_b. The contours in (a), (d), and (g) give the costs, the contours in (b), (e), and (h) give the mean response time, and the contours in (c), (f), and (i) give the standard deviation of response time. Zipf workload curves (different α) are also plotted.