Improving Reliability of Energy-Efficient Storage System

by

Shu Yin

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
May 7, 2012

Keywords: Reliability, Energy Efficient, Parallel Storage System, Modeling

Approved by

Xiao Qin, Chair, Associate Professor of Computer Science and Software Engineering
Alvin Lim, Associate Professor of Computer Science and Software Engineering
Sanjeev Baskiyar, Associate Professor of Computer Science and Software Engineering
George T. Flowers, Dean of Graduate School

Abstract

With the rapid growth of the production and storage of large scale data sets it is important to investigate methods to drive the cost of storage systems down. Many energy conservation techniques have been proposed to achieve high energy efficiency in disk systems. Unfortunately, growing evidence shows that energy-saving schemes in disk drives usually have negative impacts on storage systems. Existing reliability models are inadequate to estimate reliability of parallel disk systems equipped with energy conservation techniques. To solve this problem, we firstly propose a mathematical model - called MINT - to evaluate the reliability of a parallel disk system where energy-saving mechanisms are implemented. In this dissertation, MINT is focused on modeling the reliability impacts of two well-known energy-saving techniques - the Popular Disk Concentration technique (PDC) and the Massive Array of Idle Disks (MAID). Different from MAID and PDC which store a complete file on the same disk, the Redundancy Array of Inexpensive Disks (RAID) stripes file into several parts and stores them on different disks to ensure higher parallelism, hence higher I/O performance. However, RAID faces more challenges on energy efficiency and reliability issues. In order to evaluate the reliability of power-aware RAID, we then develop a Weibull-based model–MREED. In this dissertation, we use MREED to model the reliability impacts of a famous energy efficiency storage mechanism– the Power-Aware RAID (PARAID). Thirdly, we focus on validation of two models–MINT and MREED. It is challenging to validate the accuracy of reliability models, since we are unable to watch certain energy-efficiency systems for a couple of decades due to its time consuming and experimental costs. We introduce validated storage system simulator–DiskSim–to determine if our model and DiskSim agree with one another.

In our validation process, we compare a file access trace in a real-world file system. Last part of of this dissertation focuses on improvement of energy-efficient parallel storage systems. We propose a strategy–Disk Swapping–to improve disk reliability by alternating disks storing data that is frequently accessed with disks holding less accessed data. In this part, we focus on studying reliability improvement of PDC and MAID. At last, we further improve disk reliability by introducing multiple disk swapping strategy.

Acknowledgments

I am sincerely and heartily grateful to my advisor, Dr. Xiao Qin, for the support and guidance he showed me throughout my graduate studies at Auburn University. Dr. Qin has spent many hours guiding and mentoring me and I am sincerely grateful for the experience he has provided me as my advisor. His hard work and dedication to the academic field will serve as an important example of what a successful academic can achieve.

I would also like to thank Dr. David Umphress because he has served as my secondary advisor and I have enjoyed the feedback and encouragement that he provides. I believe that his experiences, suggestions and passions on teaching have proved to be strong influences in my early academic career. I would also like to thank Dr. Sanjeev Baskiyar and Dr. Alvin Lin for serving on my dissertation committee and providing insightful feedback when needed. I want to thank the faculty and staff of the CSSE department as they have guided me through this process. Jo-Ann Lauranitis has helped me greatly with the administrative process and I am thankful for her quick responses to all of my questions.

Our research group has been supportive and helpful all of the years I have been at Auburn University. I would like to thank Adam Manzanares because we have worked together on few projects and he has helped me meet several deadlines. Besides, his encouragements and suggestions helped me a lot for building vision of my career and the world outlook . I would also like to thank Xiaojun Ruan, Zhiyang Ding for providing me feedback and support during the dissertation process. Ziliang Zong and Kiranmai Bellam are two research group members who have helped me greatly during the first years of my stay at Auburn.I would also like to thank Jianguo Lu,

Yun Tian, James Majors, Jiong Xie and Ji Zhang for the help during the last stages of my dissertation work.

I would like to acknowledge my parents Guilin Yin and Luyun Zhu because they have served as the greatest inspiration in my life. Their climb through the journey of life is an amazing feat and I am thankful for their endless support through any situation that I have faced. I also would like to thank my best friends who offered greatly encouragements when I was depression and helped me to realize who I am and what I want.

To my beloved.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Due to current trends in computing we are facing the so called data explosion. As the use of computers to help day-to-day tasks has increased, we also face a side effect of generating large amounts of data. This data must be stored on some sort of medium and currently hard disk drives have become the most common storage medium. Large scale storage systems are being developed and installed routinely and there is a significant amount of energy that must be consumed to operate these storage systems. Many energy conservation techniques have been proposed to achieve high energy efficiency in disk systems. Unfortunately, growing evidence shows that energy-saving schemes in disk drives usually have negative impacts on storage systems. The reliability models are inadequate to estimate reliability of parallel disk systems equipped with energy conservation techniques. To solve this problem, we propose mathematical models to evaluate the reliability of parallel disk systems where energy-saving mechanisms are implemented. Furthermore, we propose a strategy to improve energy-efficient parallel disk systems reliability.

This chapter continues by developing the problem statement clearly in Section 1.1. Section 1.2 presents the scope of the research Section 1.3 summarizes the main contributions of the dissertation. Finally Section 1.4 outlines the organization of the dissertation.

1.1 Problem Statement

The number of large-scale parallel I/O systems is increasing in todays high-performance data-intensive computing systems due to the storage space required to

contain the massive amount of data. Typical examples of data-intensive applications requiring large-scale parallel I/O systems include; long running simulations [27], remote sensing applications [83] and biological sequence analysis [39]. As the size of a parallel I/O system grows, the energy consumed by the I/O system often becomes a large part of the total cost of ownership [62][91][100]. Reducing the energy costs of operating these large-scale disk I/O systems often becomes one of the most important design issues. It is known that disk systems can account for nearly 27% of the total energy consumption in a data center [37]. Even worse, the push for disk I/O systems to have larger capacities and speedier response times have driven energy consumption rates upward.

Existing energy conservation techniques can yield significant energy savings in disks. While several energy conservation schemes like cache-based energy saving approaches normally have marginal impact on disk reliability, many energy-saving schemes (e.g., dynamic power management and workload skew techniques) inevitably have noticeable adverse impacts on storage systems [12][90]. For example, dynamic power management (DPM) techniques save energy by using frequent disk spin-downs and spin-ups, which in turn can shorten disk lifetime [22][34][46], redundancy techniques [60] [102][82][89], workload skew [54][38][98], and multi-speed settings [32][76]. Unlike DPM, workload-skew techniques such as MAID [19] and PDC [58] move popular data sets to a subset of disks arrays acting as workhorses, which are kept busy in a way that other disks can be turned into the standby mode to save energy. Compared with disks storing cold data, disks archiving hot data inherently have higher risk of breaking down.

It is often challenging to improve both reliability and energy efficiency of storage systems, because little attention has been paid to evaluating reliability impacts of power management strategies on storage systems. Many excellent reliability models

have been proposed for disk systems (see, for example, [17] and [80]). However, existing disk reliability models are inadequate for evaluating reliability of disk systems epuipped with energy-saving mechanisms. For example, Shah and Elerath conducted a series of reliability analyses using field failure data of several drive models from various disk drive manufacturers [72]. Hughes and Murray investigated SATA disk drive reliability factors that bear on storage system performance [35]. They not only studied SATA drive operating failure rates, but also proposed approaches to improving reliability of storage systems comprised of multiple SATA disks [35]. Reliability models that do not consider energy-saving mechanisms are quite inaccurate when it comes to the estimation of reliability of energy-efficient disk systems. Our goal is to quantitatively investigate the reliability of parallel disk systems employing a variety energy conservation schemes using a novel mathematical model.

1.2 Research Scope

Our research focuses on models to evaluate reliability of energy-efficient parallel storage systems. We start the modeling process by capturing the behaviors of parallel disk systems coupled with power management optimization policies. Let us first make use of data access patterns as input parameters, which are used to estimate each disk's utilization and power-state transition frequency. Then, we derive each disk's reliability in terms of annual failure rate from the disk's utilization, operating temperature as well as power-state transition frequency. These parameters are key reliability-affecting factors in addition to disk ages. Finally, we calculate the reliability of the parallel disk system in accordance with the annual failure rate of each disk in the system.

This work is accomplished through the use of models and simulations. We present two models to help us model reliability of two different types of energy efficiency disk systems–ordinary disk arrays and RAID, which equipped with data striping

techniques. We model the utilization of disk serving requests and also the state transition changes and their impact on the reliability of the disk system. Using these models we developed our own simulator which we used to evaluate reliability of disk systems quickly. Our models are validated by making changes to the DiskSim simulation environment . Finally we develop a prototype implementation of a virtual file system that supports our reliability models for energy efficiency disk systems and also develop a prototype technique that improves reliability of parallel storage systems equipped with energy-saving strategies.

1.3 Contributions

The major contributions of the research presented in this dissertation follows:

1. A generic mathematical approach –MINT– to modeling reliability of energy-efficient parallel disks coupled with power management optimization policies;

2. Two reliability models for the two well-known energy-saving schemes - Popular Data Concentration scheme (PDC) and Massive Array of Idle Disks (MAID);

3. Intriguing impacts of PDC and MAID on the reliability of parallel disk systems;

4. A reliability model –MREED, which introduces Weibull analysis– is proposed for energy aware data-stripping parallel storage system;

5. Validation of the access-rate-utilization model of MREED is presented;

6. The reliability of power-aware RAID-0 and RAID-5 (PARAID-0, PARAID-5) is evaluated;

7. A prototype technique –disk swapping– is developed and implemented.

1.4 Organization

This dissertation is organized in the following manner:

Chapter 2 introduces related work that is briefly reviewed and contrasted against the contributions of this dissertation.

Chapter 3 introduces MINT model for the evaluation of disk arrays equipped with energy-saving techniques. Especially, we evaluate two well-known energy-efficient mechanisms –PDC and MAID. Thorough simulation results are also presented in this chapter.

Chapter 4 details MREED model for the evaluation of energy aware date-stripping parallel storage system. The reliability of PARAID-0 is evaluated.

Chapter 5 introduces methods for the validation of our reliabilities.

Chapter 6 presents the Disk-Swapping, which is a prototype techniques that I developed to improve reliability of parallel storage systems.

Chapter 7 summarizes the main contributions of this dissertation and presents a couple of future research directions based on the ideas contained in the dissertation.

Chapter 2

Literature Review

This chapter briefly presents previous approaches found in the literature that are most relevant to our research from two aspects: energy-efficient storage systems, and reliability impacts on disks. Fig. 2.1 shows a simplified taxonomy of storage systems.

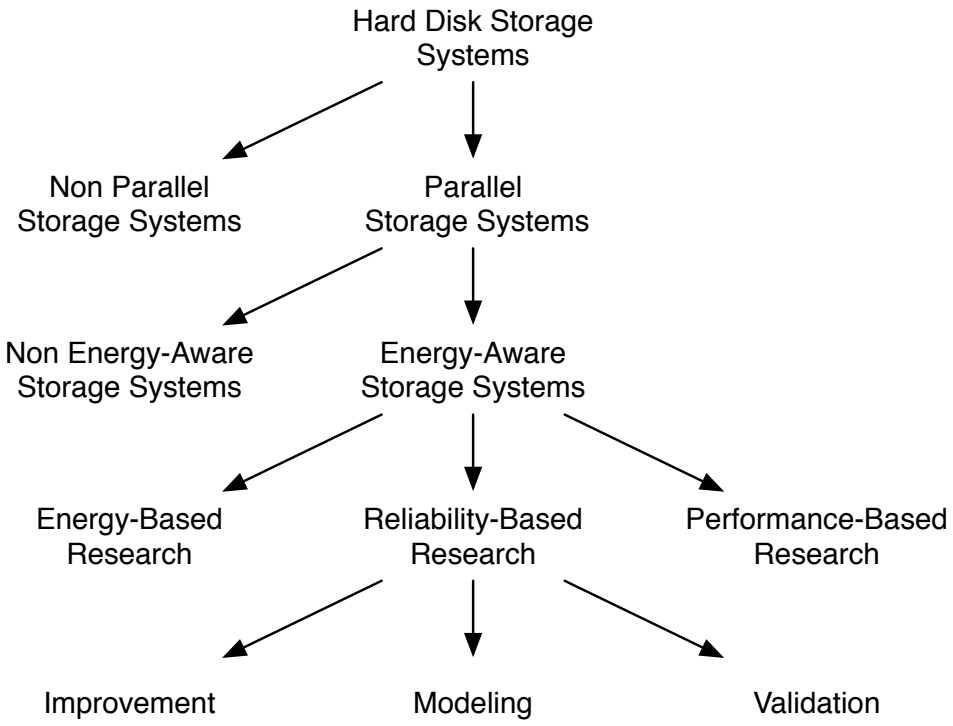

Figure 2.1: A Simplified Taxonomy of Storage Systems Research

2.1 Hard Disk Drive Storage Systems

Introduced by IBM in 1956, a hard disk drive (HDD) is a device for storing and retrieving digital information. Hard disk drives have been a dominant device

for secondary storage of data in general purpose computers since the early 1960s. Hard drives have maintained this position because advances in their recording capacity, cost, reliability, and speed have kept pace with the requirements for secondary storage [51].

The capacity of hard drives has grown exponentially over time. When hard drives became available for personal computers (PCs), they only offered five-megabytes capacity. During the mid-1990s, the typical hard disk drive for a PC had a capacity of about one-gigabyte [1]. In the year 2007, Hitachi firstly introduced the world's one-terabyte hard disk drive [5]. As of January 2012, desktop hard disk drives typically had a capacity of 500 to 2000 gigabytes, while the largest-capacity drives were four terabytes [8].

The latency of a disk access can therefore be broken down into three main aspects: seek, rotational and transfer latencies. Seek latency refers to the time it takes to position the read/write head over the proper track. The seek process involves a mechanical transitional movement that may require an acceleration in the beginning and a deceleration and a repositioning in the end. As a result, although disk seek times have been reduced, short seek times have not kept up with the rates of improvement of silicon processors. While processing rates have improved by more than an order of magnitude, average seek times have shrunk to only half of their values of a decade ago [11].

Rotational latency, which is delay for the rotation of a disk to bring the required disk sector under the read-write mechanism. This characteristic is mainly relies on rotational speed of a disk, measured in revolutions per minute (RPM). Due to electronic , mechanical as well as the manufacturing constraints, it is hard to shorten the latency by increasing the rotational speed of disks. The RPM of a disk have tripled in the past decades; the fastest hard disk drive was produced by Seagate in 2000 with

RPM 15000 [3]. A study shows that it is unlikely that there will be a disk rotational speed increase in the near future [29].

The third type of delay is transfer time, which is the time for target sectors to pass under a read/write head. Disk transfer times are determined by the rotational speed and storage density (in bytes/square inch). Disk areal densities continue to increase at 50 to 55% per year, leading to dramatic increases in sustained transfer rates, averaged at 40% per year [30].

The disk performance has been steadily improving with more pronounced gains for large transfer access time. The maximum sustained bandwidth (MB/s) is roughly proportional to the linear density. The compound annual growth rate (CAGR) of bandwidth kept around 20% from the year 1996 to early 2002 and; recently, it is more likely to fall within the range of 10 to 15%. Currently a high performance disk drive would have a maximum sustained bandwidth of approximately 171 MB/s [47].

2.2 Parallel Storage Systems

A single disk storage system is out of its reach in terms of scientific computation, because it often requires significant computational power and involves large amount of data. Advances in communications technology allow numbers of effectively unbounded processing power and storage capacity to be used to solve much larger problems than those that only handled by single machine.

RAID is an example of advanced storage technique first introduced by David Partterson, Garth A. Gibson, and Randy Katz at the University of California, Berkeley in 1987 [56]. The different schemes or architectures are named by the term RAID followed by a number (e.g., RAID-0, RAID-1). Each scheme provides a different balance between two key goals: to increase date reliability and to increase read/write performance. Mainly, there are three RAID levels; many more variations have been proposed.

- RAID-0 (block-level striping without parity or mirroring) has no (or zero) redundancy. It provides improved performance and additional storage without fault tolerance. Hence simple stripe sets are normally referred to as RAID 0. Any drive failure destroys the array, and the likelihood of failure increases with more drives in the array [69][41].

- RAID 1 (mirroring without parity or striping), data is written identically to multiple drives, thereby producing a "mirrored set"; at least two drives are required to constitute such an array. While more constituent drives may be employed, many implementations deal with a maximum of only two. Of course, it might be possible to use such a limited level 1 RAID to effectively mask the limitation [45][74][31].

- RAID 5 (block-level striping with distributed parity) distributes parity along with the data and requires all drives but one to be present to operate; data in the array will not lost even in case of a single drive failure. Upon drive failure, any subsequent reads can be calculated from the distributed parity such that failed the drive can be rebuild by the end user. However, a single drive failure results in reduced performance of the entire array until the failed drive has been replaced and the associated data reconstructed [52][14]. RAID 5 requires at least three disks.

The Parallel Virtual File System (PVFS) is an open source parallel file system. A parallel file system is a type of distributed file system that distributes file data across multiple servers and provides for concurrent access by multiple tasks of a parallel application. PVFS was designed for large-scale cluster computing systems. PVFS focuses on high performance access to large data sets. It consists of a server process and a client library, both of which are written entirely of user-level code [33][77][96].

Lustre is another parallel distributed file system, generally used for large scale cluster computing. The name Lustre is a portmanteau word derived from Linux and cluster. Because Lustre has high performance capabilities and open licensing, it is often deployed in super computers [57][28][15]. At the present time, fifteen of the top 30 supercomputers in the world have Lustre file systems installed, including the world's fastest TOP500 supercomputer [9], K computer [7].

Ceph is a free software distributed file system initially created by Sage Weil [86]. Ceph's main goals are to be POSIX-compatible, and completely distributed without a single point of failure. The data is seamlessly replicated, making Ceph fault tolerant [43].

PanFS is a parallel distributed file system developed by Pansas, INC. It creates a single pool of storage under a global namespace that provides the ability to support multiple applications and workflows in a single storage system with optimal performance for complex technical applications. PanFS eliminates the need for multiple islands of storage [18][53][88].

2.3 Energy-Efficient Parallel Disk Systems.

Hard disk drives (HDD) are made up of various electrical, electronic, and mechanical components [97]. An array of techniques were developed to save energy in single HDDs. Energy dissipation in disk drives can be reduced at the I/O level (e.g., dynamic power management [23][46] and multi-speed disks [34]), the operating system level (e.g., power-aware caching/prefetching [102][76]), and the application level (e.g. software DMP [75] and cooperative I/O [87]).

Existing energy-saving techniques for parallel disk systems often rely on one of the two basic ideas - power management and workload skew. Power management schemes conserve energy by turning disks into standby after a period of idle time. Although multi-speed disks are not widely adopted in storage systems, power management

has been successfully extended to address the energy-saving issues in multi-speed disks [34][32][42]. The basic idea of workload skew is to concentrate I/O workloads from a large number of parallel disks into a small subset of disks allowing other disks to be placed in the standby mode [58][19][66][59].

2.4 Reliability Impacts of Power Management on Disks.

Recent studies show that both power management and workload skew schemes inherently impose adverse reliability impacts on disk systems [12][90]. For example, the power management schemes are likely to result in a huge number of disk spin-downs and spin-ups that can significantly reduce the lifespan of hard disks.

The workload skew techniques dynamically migrates frequently accessed data to a subset of disks [65] [49], which inherently have higher risk of breaking down than other disks usually being kept standby. Disks storing popular data tend to have high failure rates due to extremely unbalanced workload. Thus, the popular data disks have a strong likelihood to become a reliability bottleneck. The design of our MINT, presented in this dissertation, is orthogonal to the aforementioned energy saving studies (see Section 3.2), because MINT is focused on reliability impacts of the power management and workload skew schemes in parallel disks.

2.5 Reliability Models of Disk Systems.

A malfunction of any components in a hard disk drive could lead to a failure of the disk. Reliability - one of the key characteristics of disks - can be measured in terms of mean-time-between-failure (MTBF). Disk manufacturers usually investigate MTBFs of disks either by laboratory testing or mathematical modeling. Although disk drive manufacturers claim that MTBF of most disks is more than 1 million hours [71], users have experienced a much lower MTBF from their field data [25]. More importantly, it is challenging to measure MTBF because of a wide range of contributing

factors including disk age, utilization, temperature, and power-state transition frequency [36][24][63].

A handful of reliability models have been successfully developed for storage systems. For example, Pâris *et. al* investigated an approach to computing both average failure rate and mean time to failure in distributed storage systems [55]; Elerath and Pecht proposed a flexible model for estimating reliability of RAID storage [26]; and Xin *et. al* developed a model to study disk infant mortality [93]. Unlike these reliability models tailored for conventional parallel and distributed disk systems, our MINT model proposed in Chapter 3 pays special attention to reliability of parallel disk systems coupled with energy-saving mechanisms.

2.6 Validation of Models.

Model validation means substantiation that a computerized model within its domain of applicability prossess a satisfactory range of accuracy consistent with the intended application of the model [70]. Major ways to validate models include Historical Methodes, extreme condition test, and Comparison to Other Models [67][13][48]. For example, R.E. Brown *et. al* validated their distributions system reliability models by adjusting default component reliability parameters so that predicted results match historical results. [16]. In Extreme Condition Tests, the model structure and outputs should be plausible for any extreme and unlikely combination of levels of factors in the system. We developed a trace-driven simulation model using the Berkeley Web Trace [2] as a reference model to compare with our MINT model for the validation purpose. The major reason that we used a Web trace is that our research pays more attention to read-intensive I/O activities and Web workloads impose higher read load than write load [64][79][44].

2.7 Reliability Improvements

Storage clusters consisting of thousands of disk drives are widely employed because of their large capacity and high I/O throughput. However, the reliability of large storage clusters is far worse than that of smaller storage systems due to the increased number of storage nodes. RAID technology is no longer sufficient to guarantee high data reliability for large-scale storage cluster systems, because disk rebuild time lengthens as disk capacity grows [95]. Researchers developed various methods to improve reliability of storage clusters. For example, Xie *et. al* developed a novel data reconstruction strategy, called multi-level caching-based reconstruction optimization (MICRO), which can be applied to RAID-structured mobile storage systems. MICRO can noticeably shorten reconstruction times and user response times while saving energy [92]; Xin *et. al* presented fast recovery mechanism (FARM), a distributed recovery approach that exploits excess disk capacity and reduces data recovery time [94]; Zhang *et. al* proposed a fast and efficient reverse lookup scheme named Group-based Shifted Declustering (G-SD) layout that is able to locate the whole content of the failed node [101].

Chapter 3

MINT: A Reliability Modeling Framework for Energy-Efficient Parallel Disk Systems

Many energy conservation techniques have been proposed to achieve high energy efficiency in disk systems. Unfortunately, growing evidence shows that energy-saving schemes in disk drives usually have negative impacts on storage systems. Existing reliability models are inadequate to estimate reliability of parallel disk systems equipped with energy conservation techniques. To solve this problem, we propose a mathematical model - called MINT - to evaluate the reliability of a parallel disk system where energy-saving mechanisms are implemented. In this paper, we focus on modeling the reliability impacts of two well-known energy-saving techniques - the Popular Disk Concentration technique (PDC) and the Massive Array of Idle Disks (MAID). We started this research by investigating how PDC and MAID affect the utilization and power-state transition frequency of each disk in a parallel disk system. We then model the annual failure rate of each disk as a function of the disk's utilization, power-state transition frequency as well as operating temperature, because these parameters are key reliability-affecting factors in addition to disk ages. Next, the reliability of a parallel disk system can be derived from the annual failure rate of each disk in the parallel disk system. Finally, we used MINT to study the reliability of a parallel disk system equipped with the PDC and MAID techniques. Experimental results show that PDC is more reliable than MAID when disk workload is low. In contrast, the reliability of MAID is higher than that of PDC under relatively high I/O load.

3.1 Motivations

Parallel disk systems, providing high-performance data-processing capacity, are of great value to large-scale parallel computers [4]. A parallel disk system comprised of an array of independent disks can be built from low-cost commodity hardware components. In the past few decades, parallel disk systems have increasingly become popular for data-intensive applications running on massively parallel computing platforms [81].

Existing energy conservation techniques can yield significant energy savings in disks. While several energy conservation schemes like cache-based energy saving approaches normally have marginal impact on disk reliability, many energy-saving schemes (e.g., dynamic power management and workload skew techniques) inevitably have noticeable adverse impacts on storage systems [12][90]. For example, dynamic power management (DPM) techniques save energy by using frequent disk spin-downs and spin-ups, which in turn can shorten disk lifetime [22] [34] [46], redundancy techniques [60] [102] [82] [89], workload skew [54] [38] [98], and multi-speed settings [32] [76]. Unlike DPM, workload-skew techniques such as MAID [19] and PDC [58] move popular data sets to a subset of disks arrays acting as workhorses, which are kept busy in a way that other disks can be turned into the standby mode to save energy. Compared with disks storing cold data, disks archiving hot data inherently have higher risk of breaking down.

Unfortunately, it is often difficult for storage researchers to improve reliability of energy-efficient disk systems. One of the main reasons lies in the challenge that every disk energy-saving research faces today, how to evaluate reliability impacts of power management strategies on disk systems. Although reliability of disk systems can be estimated by simulating the behaviors of energy-saving algorithms, there is lack of fast and accurate methodology to evaluate reliability of modern storage systems with high-energy efficiency. To address this problem, we developed a mathematical

reliability model called MINT to estimate the reliability of a parallel disk system that employs a variety of reliability-affecting energy conservation techniques [99].

In this paper, we first study the reliability of a parallel disk system equipped with a well-known energy-saving scheme— the MAID [19] technique. I/O load skewing techniques like MAID inherently affect reliability of parallel disks because of two reasons: First, disks storing popular data tend to have high I/O utilization than disks storing cold data. Second, disks with higher utilization are likely to have higher risk of breaking down. To address the adverse impact of load skewing techniques on disk reliability, a disk swapping strategy was proposed to improve disk reliability in MAID by switching the roles of data disks and cache disks. We evaluate impacts of the disk swapping scheme on the reliability of MAID-based parallel disk systems.

In this paper, our contributions are as follows:

1. We studied a model for Massive Array of Idle Disks (MAID) based on Mathematical Reliability Models for Energy-efficient Parallel Disk System (MINT) [99];

2. We built single disk swapping and multiple disk swapping mechanisms to improve reliability of various load skewing techniques.

3. We studied the impacts of the disk swapping schemes on the reliability of MAID.

The remainder of this paper is organized as follows. Section 3.2 outlines the design and implementation of the MINT reliability modeling framework, which relies on disk utilization, temperature, and power-state transition frequency. Section 3.3 presents reliability models for MAID and PDC schemes along with the preliminary results.

3.2 The MINT Reliability Model

3.2.1 Framework

Fig. 3.1 depicts the framework of the MINT reliability model for parallel disk systems with energy conservation schemes. MINT is composed of a single disk reliability model, a system-level reliability model, and three reliability-affecting factors - temperature, disk state transition frequency (hereinafter referred to as frequency) and utilization. Many energy-saving schemes (e.g., PDC [58] and MAID [19]) inherently affect reliability-related factors like disk utilization and transition frequency. Given an energy optimization mechanism, MINT first transfers data access patterns into the two reliability-affecting factors - frequency and utilization. The single-disk reliability model can derive individual disk's annual failure rate from utilization, power-state transition frequency, age, and temperature. Each disk's reliability is used as input to the system-level reliability model that estimates the annual failure rate of parallel disk systems.

For simplicity without losing generality, we consider four reliability-related factors in MINT. This assumption does not necessarily indicate that disk utilization, age, temperature, and power-state transitions are the only parameters affecting disk reliability. Other factors having impacts on reliability include handling, humidity, voltage variation, vintage, duty cycle, and altitude [25]. If a new factor has to be integrated into MINT, we can extend the single reliability model described in Section 3.2.5. Since the infant mortality phenomena is out the scope of this study, we pay attention to disks that are more than one year old.

3.2.2 Impacts of Utilization on Disk Annual Failure Rate

Disk utilization can be characterized as the fraction of active time of a disk drive out of its total powered-on-time [61]. In our single disk reliability model, the impacts

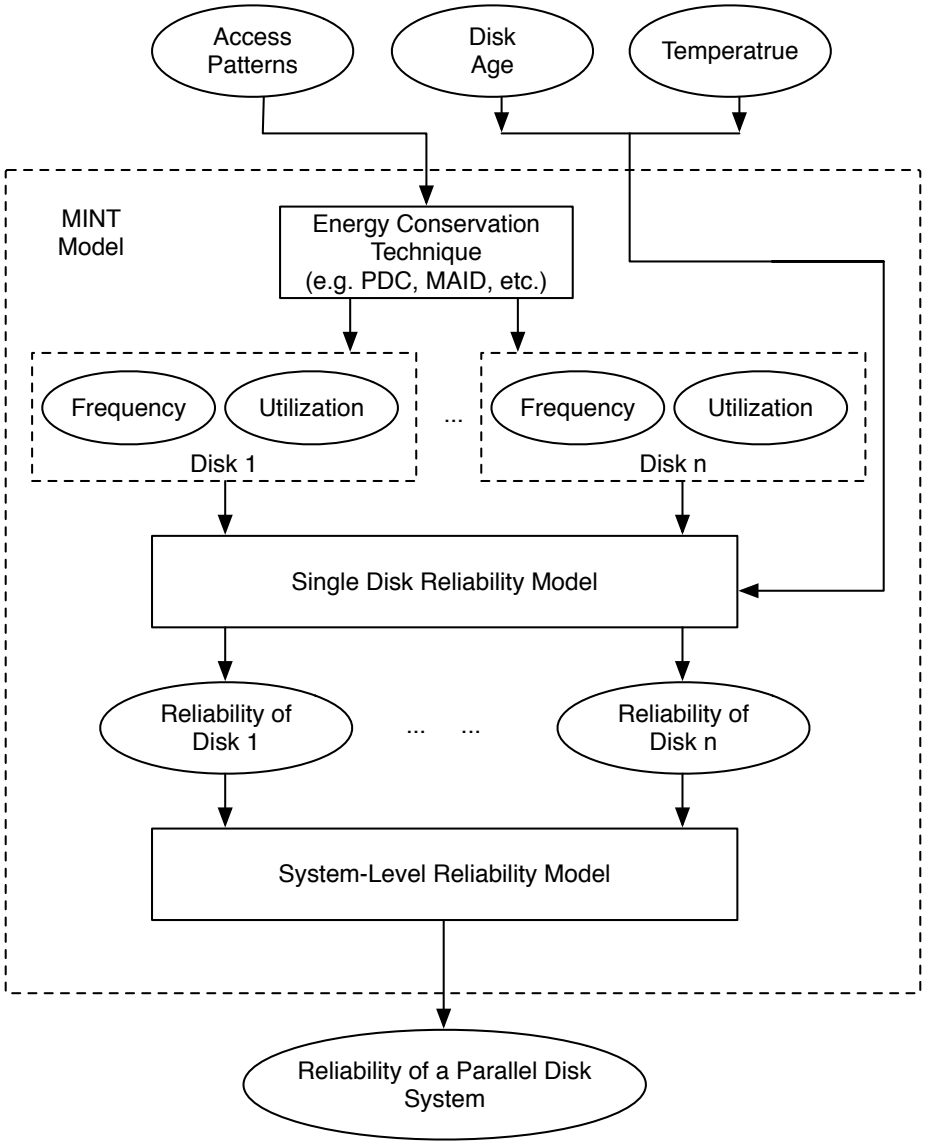

Figure 3.1: The Framework of the MINT Reliability Model

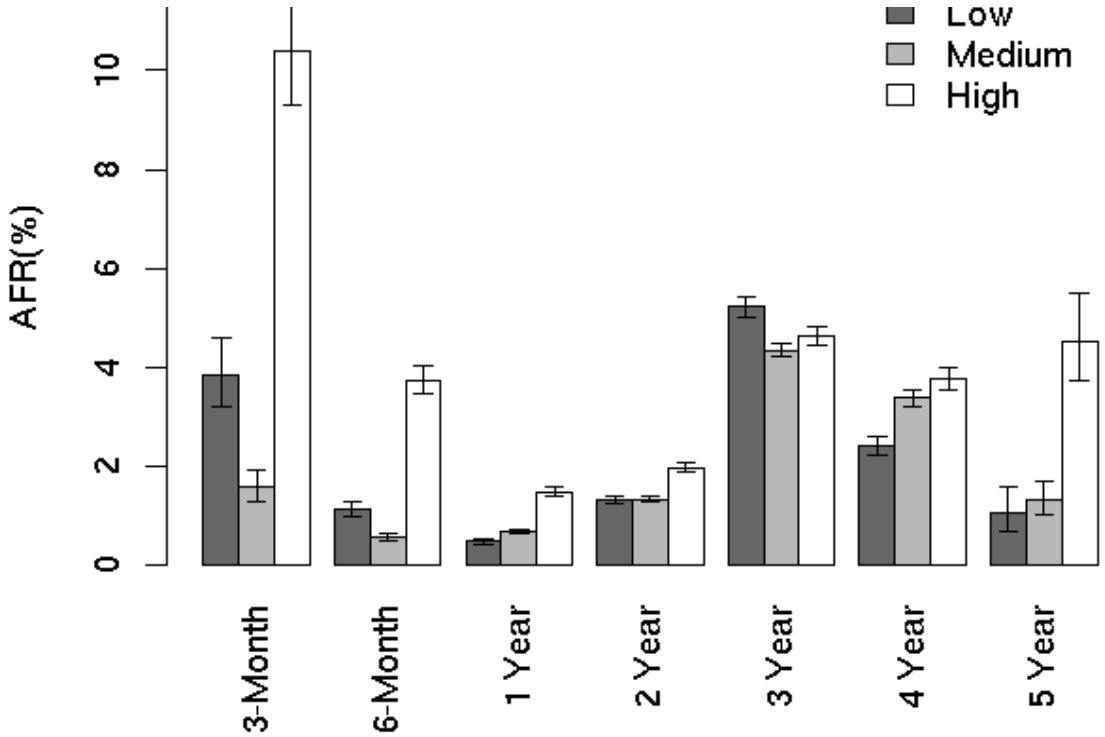

Figure 3.2: Utilization Impacts on AFR (by Google)

of disk utilization on reliability is good way of providing a baseline characterization of disk annual failure rate (AFR). Using field failure data collected by Google, shows the impact of utilization on AFR across the different age groups. Pinheiro *et al.* studied the impact of utilization on AFR accross different disk age groups. Pinheiro *et al.* categorized disk utilization in three levels - low, medium, and high. Fig. 3.2 shows AFRs of disks whose ages are 3 months, 6 months, 1 year, 2 years, 3 years, 4 years, and 5 years under the three utilization levels. Since the single-disk reliability model needs a baseline AFR derived from a numerical value of utilization, we make use of the polynomial curve-fitting technique to model the baseline value of a single disk's AFR as a function of utilization. Thus, the baseline value (i.e., $BaseValue$ in Eq. 4.1) of AFR for a disk can be calculated from the disk's utilization. For example, Fig. 3.3 shows the AFR value of a 3-year old disk as a function of its utilization. The curve plotted in Fig. 3.3 can be modeled as a utilization-reliability function described as Eq. 3.1 below:

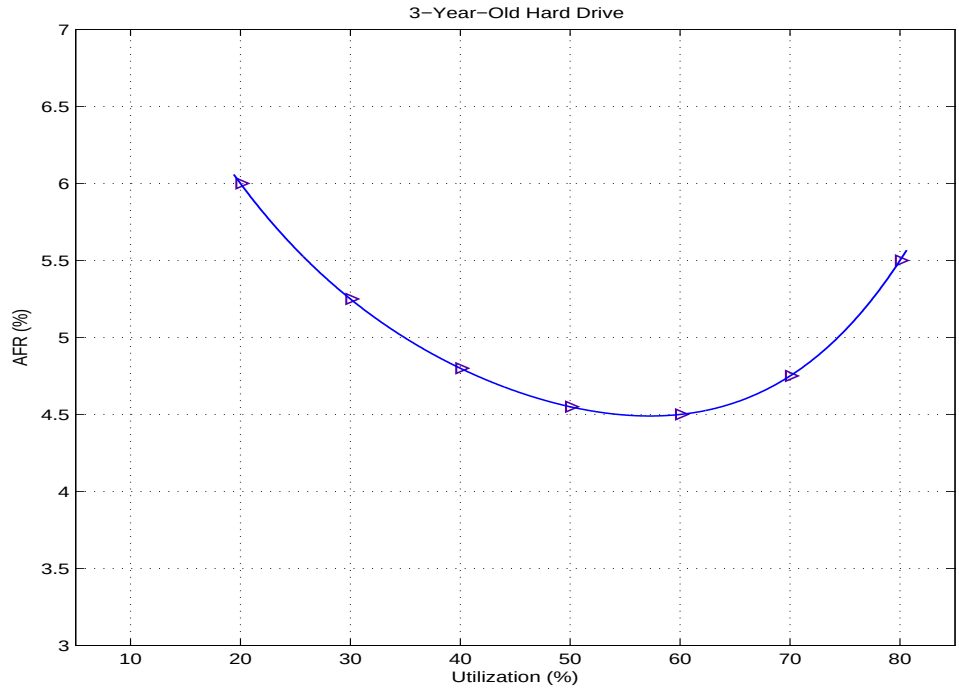

Figure 3.3: 3-Year-Old HDD Utilization Impacts on AFR

$$\begin{aligned} R(u) =& 4.167e^{-7}u^4 - 7.5e^{-5}u^3 + 5.968e^{-3}u^2 - \\ & - 2.575e^{-1}u + 9.3, \quad \text{for all } u \in [0, 100] \end{aligned} \tag{3.1}$$

where $R(u)$ represents the AFR value as a function of a certain disk utilization u. With Eq. 3.1 in place, one can readily derive annual failure rate of a disk if its age and utilization are given. For example, for a 3-year old disk with 50% utilization (i.e., $u = 50\%$), we can obtain the AFR value of this disk as $R(u) = 4.8\%$. Fig. 3.3 suggests that unlike the conclusions drawn in a previous study (see [78]), a low disk utilization does not necessarily lead to low AFR. For instance, given a 3-year old disk, the AFR value under 30% utilization is even higher than AFR under 80% utilization.

3.2.3 Impacts of Temperature on Disk Annual Failure Rate

Temperature is often considered as the most important environmental factor affecting disk reliability. Field failure data of disks in a Google data center (see Fig. 3.4) shows that in most cases when temperatures are higher than 35°C, increasing temperatures lead to an increase in disk annual failure rates. On the other hand, Fig. 3.4 indicates that in the low and middle temperature ranges, the failure rates decreases when temperature increases [61].

Growing evidence shows that disk reliability should reflect disk drives operating under environmental conditions like temperature [25]. Since temperature (e.g., meaured 1/2" from the case) apparently affect disk reliability, the temperature can be considered as a multiplier (hereinafter referred to as temperature factor) to baseline failure rates where environmental factors are integrted [25]. Given a temperature, one must decide the corresponding temperature factor (see $TemperatureFactor$ in Eq. 4.1) that can be multiplied to the base failure rates. Using Google's field failure data plotted in 3.4, we attempted to calculate temperature factors under various temperatures ranges for disks with different ages. More specifically, Fig. 3.4 shows annual failure rates of disks whose ages are from 3-month to 4-year old. For disk drives whose ages fall in each age range, we model the temperature factor as a function of drive temperature. Thus, six temperature-factor functions must be derived.

Now we explain how to determine a temperature facotr for each temperature under each age range. Let us choose 25°C as the base temperature value, because room temperatures of data centers in many cases are set as 25°C controlled by cooling systems. Thus, the temperature factor is 1 when temperature is set to the base temperature - 25°C. Let T denote the average temperature, we define the temperature factor for temperature T as $T/25$ if T is larger than 25°C. When T exceeds 45°C, the temperature factor becomes a constant (i.e., $1.8 = 45/25$). Due to space limit, we only show how temperature affects the temperature factor of a 3-year old disk in

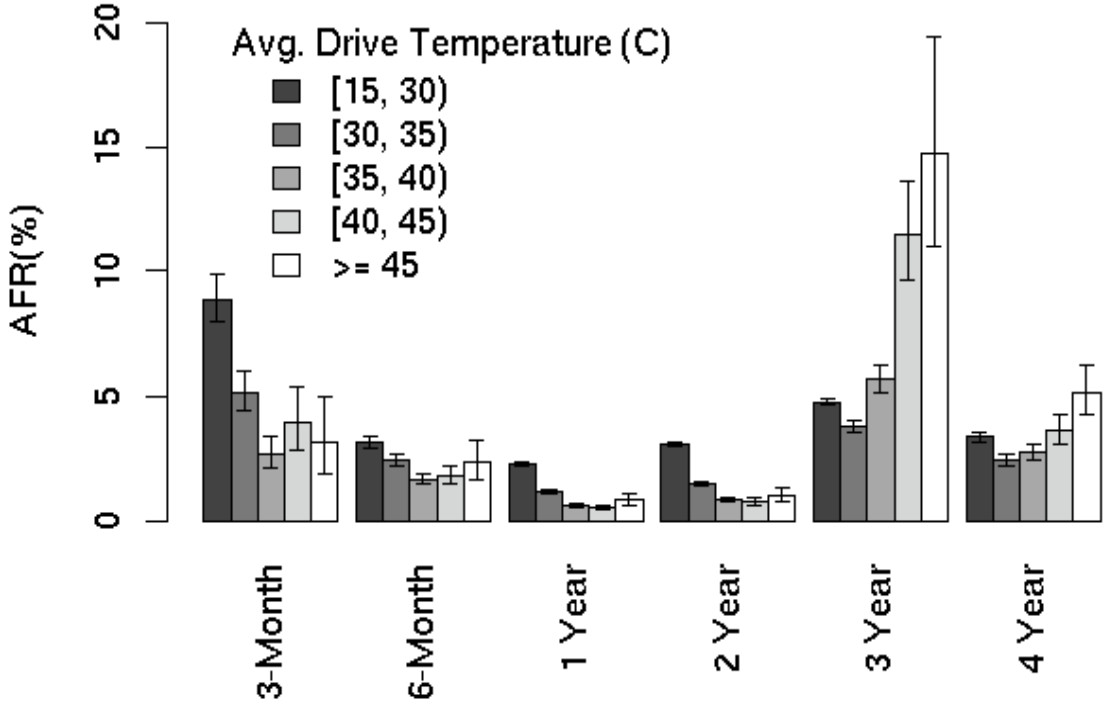

Figure 3.4: Average Drive Temperature Impacts on AFR (by Google)

Fig. 3.4. Note that the temperature-factor functions for disks in other age ranges can be modeled in a similar way. Fig. 3.5 shows the temperature-factor function derived from Fig. 3.4 for 3-year old disks. We can observe from Fig. 3.4 that AFRs increase to 215% of the base value when the temperature is between 40 to 45°C.

3.2.4 Power-State Transition Frequency

To conserve energy in single disks, power management policies turn idle disks from the active state into standby. The disk power-state transition frequency (or frequency for short) is often measured as the number of power-state transitions (i.e., from active to standby or vice versa) per month. The reliability of an individual disk is affected by power-state transitions and; therefore, the increase in failure rate as a function of power-state transition frequency has to be added to a baseline failure rate (see Eq. 4.1 in Section 3.2.5). We define an increase in AFR due to power-state

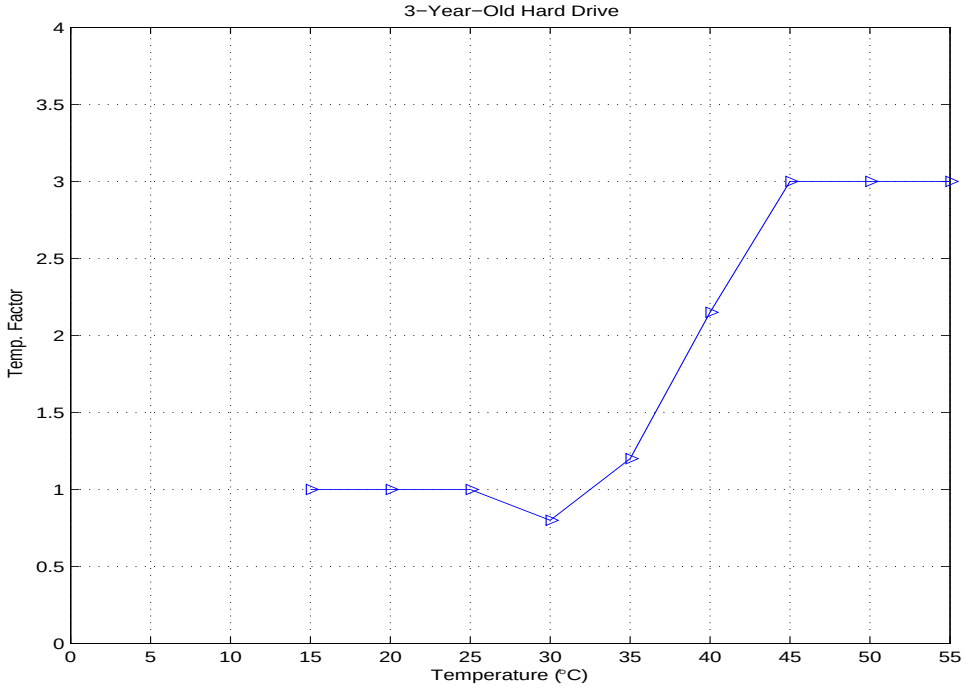

Figure 3.5: Temperature-Factor Function of 3-Year-Old HDDs

transitions as power-state transition frequency adder (frequency adder for short). The frequency adder is modeled by combining the disk spindle start/stop failure rate adders described by IDEMA [78] and the PRESS model [90]. Again, we focus on 3-year old disk drives. Fig. 3.6 demonstrates frequency adder values as a function of power-state transition frequency. Fig. 3.6 shows that high frequency leads to a high frequency adder to be added into the base AFR value. We used the quadratic curve fitting technique to model the frequency adder function (see Eq. (4.2)) plotted in Fig. 3.6.

$$R(f) = 1.51e^{-6}f^2 - 1.09e^{-5}f + 1.39e^{-2}, f \in [0, 100] \tag{3.2}$$

where f is a power-state transition frequency, $R(f)$ represents an adder to the base AFR value. For example, suppose the transition frequency is 300 per month, the base AFR value needs to be increase by 1.33%.

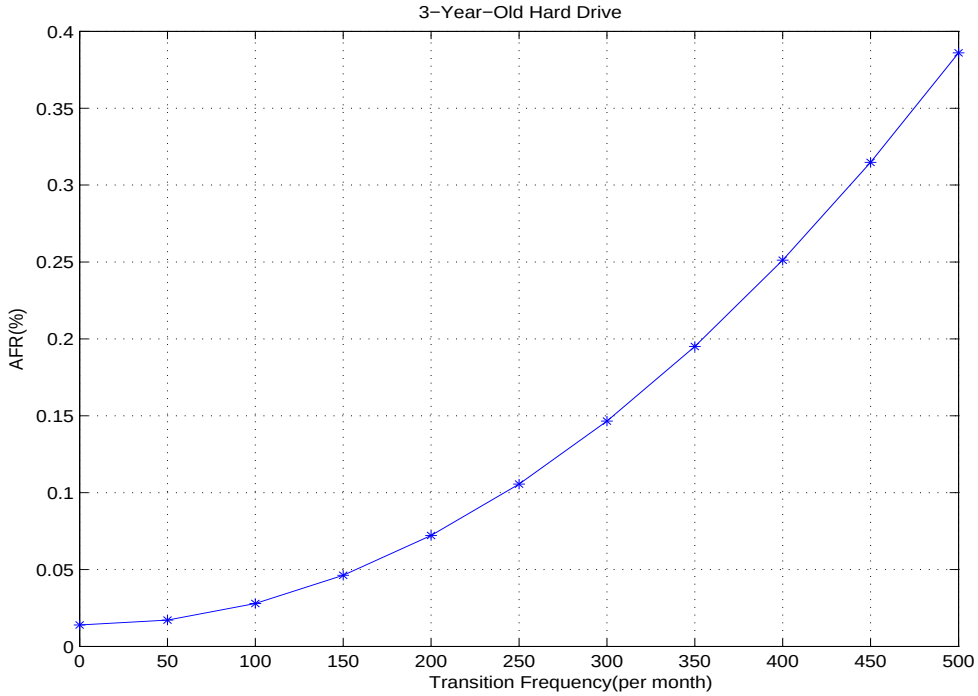

Figure 3.6: Impacts of Transition Frequency on Frequency adder of 3-Year-Old HDDs

3.2.5 Single Disk Reliability Model

Single-disk reliability can not be accurately described by one valued parameter, because the disk drive reliability is affected by multiple factors (see Sections 3.2.2,3.2.3, and 3.2.4). Though recent studies attempted to consider multiple reliability factors (see, for example, PRESS [90]), few of prior studies investigated the details of combining the multiple reliability factors. We model the single-disk reliability in terms of annual failure rate (AFR) in the following three steps. We first compute a baseline AFR as a function of disk utilization. We then use temperature factor as a multiplier to the baseline AFR. Finally, we add a power-state transition frequency adder to the baseline value of AFR. Hence, the failure rate R of an individual disk

can be expressed as:

$$\begin{aligned} R = & \alpha \times BaseValue \times TemperatureFactor + \\ & + \beta \times FrequencyAdder \end{aligned} \tag{3.3}$$

where $BaseValue$ is the baseline failure rate derived from disk utilization (see Section 3.2.2), $TemperatureFactor$ is the temperature factor (or temperature multiplier; see Section 3.2.3), $FrequencyAdder$ is the power-state transition frequency adder to the base AFR (see Section 3.2.4), and α and β are two coefficients to reliability R. If reliability R is more sensitive to frequency than to utilization and temperature, then β must be greater than α. Otherwise, β is smaller than α. In either cases, α and β can be set in accordance with R's sensitivities to utilization, temperature, and frequency. In our experiments, we assume that all the three reliability-related factors are equally important (i.e., α=β=1). Ideally, extensive field tests allow us to analyze and test the two coefficients. Although α and β are not fully evaluated by field testing, reliability results are valid because of the following two reasons: first, we have used the same values of α and β to evaluate impacts of the two energy-saving schemes on disk reliability (see Section 3.3.1); second, the failure-rate trend of a disk when α and β are set to 1 are very similar to those of the same disk when the values of α and β do not equal to 1.

With Eq. 4.1 in place, we can analyze a disk's reliability in turns of annual failure rate (AFR). Fig. 3.7 shows AFR of a three-year-old disk when its utilization is in the range between 20% and 80%. We observe from Fig. 3.7 that increasing temperature from 35°C to 40°C gives rise to a significant increase in AFR. Unlike temperature, power-state transition frequency in the range of a few hundreds per month has marginal impact on AFR. It is expected that when transition frequency is extremely high, AFR becomes more sensitive to frequency than to temperature.

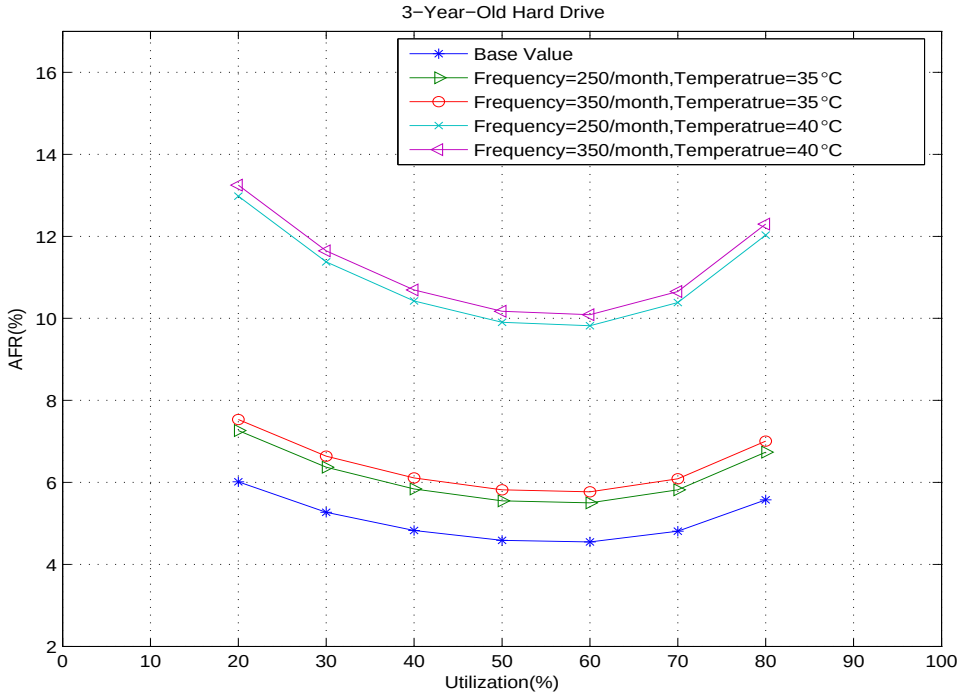

Figure 3.7: 3-Year-Old HDD Combined Factors Impacts on AFR (Single Disk Reliability Model)

3.3 Reliability Models for MAID and PDC

3.3.1 MAID- Massive Array of Idle Disks

Background

The MAID (Massive Arrays of Idle Disks) technique - developed by Colarelli and Grunwald - aims to reduce energy consumption of large disk arrays while maintaining acceptable I/O performance [19]. MAID relies on data temporal locality to place replicas of active files on a subset of cache disks, thereby allowing other disks to spin down. Fig. 3.8 shows that MAID maintains two types of disks - cache disks and data disks. Popular files are copied from data disks into cache disks, where the LRU policy is implemented to manage data replacement in cache disks. Replaced data is discarded by a cache disk if the data is clean; dirty data has to be written back to the corresponding data disk. To prevent cache disk from being overly loaded, MAID

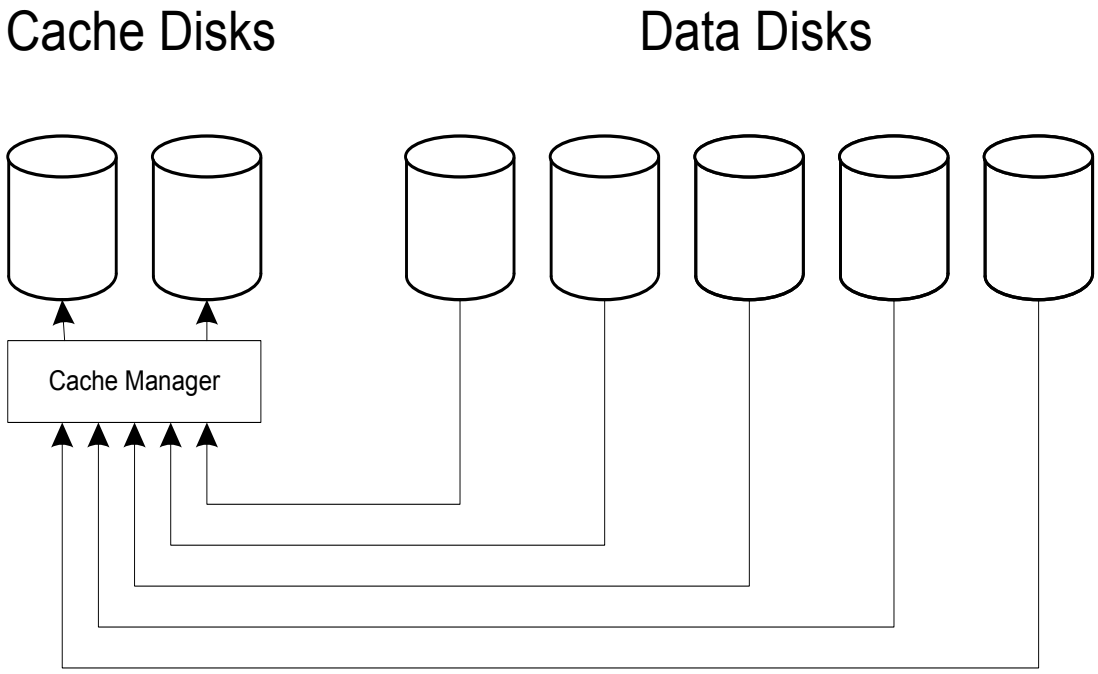

Figure 3.8: MAID System Structure

avoids copying data to cache disks that have reached their maximum bandwidth. Three components integrated in the MAID model include: (1) a power management policy that switches idle disks into the standby mode if the disks are sitting idle for a certain period of time; (2) a data placement module that either linearly places successive blocks on a disk drive or uniformly distributes data blocks across multiple drives; (3) a cache disk controller that determines the number of disks performing as cache disks [19].

Modeling Utilization of Disks in MAID

Recall that the annual failure rate of each disk can be calculated using disk age, utilization, operating temperature as well as power-state transition frequency. To model reliability of a disk array equipped with MAID, we have to first address the issue of modeling disk utilization used to calculate base annual failure rates. In this subsection, we develop a utilization model capturing behaviors of a MAID-based disk

array. The utilization model takes file access patterns as an input and calculates the utilization of each disk in the disk array.

Disk utilization is computed as the fraction of active time of a disk drive out of its total powered-on-time. Now we describe a generic way of modeling the utilization of a disk drive. Let us consider a sequence of I/O accesses with N I/O phases. We denote T_i as the length or duration of the ith I/O phase. Without loss of generality, we assume that a file access pattern in an I/O phase remains unchanged. The file access pattern, however, may vary in different phases. The relative length or weight of the ith phase is expressed as $W_i = T_i/T$ where $T = \sum_{i=1}^{N} T_i$ is the total length of all the I/O phases. Suppose the utilization of a disk in the ith phase is ρ_i, we can write the overall utilization ρ of the disk as the weighted sum of the utilization in all the I/O phases. Thus, we have

$$\rho = \sum_{i=1}^{N} (W_i \times \rho_i) = \sum_{i=1}^{N} (\frac{T_i}{T} \times \rho_i) \tag{3.4}$$

Let $F_i = (f_{i1}, f_{i2}, ..., f_{in_i})$ be a set of n_i files residing in the disk in the ith phase. The utilization ρ_i (see Eq. 4.5) of the disk in phase i is contributed by I/O accesses to each file in set F_i. Thus, ρ_i in Eq. 4.5 can be written as:

$$\rho_i = \sum_{j=1}^{n_i} (\lambda_{ij} \times s_{ij}) \tag{3.5}$$

where λ_{ij} is the file access rate of file f_{ij} in F_i and s_{ij} is the mean service time of file f_{ij}. Note that I/O accesses to each file in set F_i are modeled as a Poisson process; file access rate and service time in each phase i are given a priori. We assume that there are n hard drives with k phases. In the l-th phase, let f_{ijl} be the j-th file on

the i-th disk, where $i \in (1, 2, \cdots, n)$, $j \in (1, 2, \cdots, m_i)$, $l \in (1, 2, \cdots k)$. We have:

$$\begin{aligned} F_{1l} &= \{ f_{11l}, f_{12l}, \cdots, f_{1m_1l} \} \\ &\vdots \\ F_{nl} &= \{ f_{n1l}, f_{n2l}, \cdots f_{nm_nl} \} \end{aligned} \tag{3.6}$$

where m_i is the number of files on the ith disk and F_{il} is the total files on the same disk. Since frequently accessed files are duplicated to cache disks, we model below an updated file placement after copying the frequently accessed files.

$$\begin{aligned} F'_{1l} &= \left\{ f'_{11l}, f'_{12l}, \cdots, f'_{1m'_1l} \right\} \\ &\vdots \\ F'_{nl} &= \left\{ f'_{n1l}, f'_{n2l}, \cdots, f'_{nm'_nl} \right\} \end{aligned} \tag{3.7}$$

where $m'_i$is the number of the files on the i-th disk, f'_{ijl} is the j-th file at the l-th phase and F'_{il} is the set of files on the same disk after the files are copied. We can calculate the utilization for jth file in the lth phase on the ith disk as $\rho_{ijl} = \lambda_{ijl} \times t$. We assume that $\rho_{i1l} \geq \rho_{i2l} \geq \cdots \geq \rho_{im_1l}$, meaning that files are placed in a descending order of utilization. After the frequently accessed files are copied to the cache disks, we denote the updated utilization contributed by files including copied ones as $\rho'_{i1l} \geq \rho'_{i2l} \geq \cdots \geq \rho'_{im_1l}$. It is intuitive that the utilization of disk i should be smaller than 1. When a disk reaches its maximum utilization, the disk also reaches its maximum bandwidth denoted as B_i. For both cache and data disks, we express the utilization

for ith disk in phase l as:

$$\begin{aligned}\rho'_{il} &= \frac{I/O time + Copying\ time}{T} \\ &= \sum_{j=1}^{m'_i} \rho'_{ijl} + \frac{Copying\ time}{T}\end{aligned} \tag{3.8}$$

where T is the time interval of the lth I/O phase. The first and second items on the bottom-line on the right-hand side of Eq. 3.8 are the utilizations caused by accessing files and duplicating files from data disks to cache disks, respectively.

Since files on cache disks are duplicated from data disks, frequently accessed files must be copied from data disks and written down to cache disks. As such, we must consider disk utilization incurred by the data duplication process. To quantify utilization overhead caused by data replicas, we define a set $F_{il}^{M\_out}$ of files copied from the ith data disk to cache disks in phase l. Similarly, we define a set $F_{il}^{M\_in}$ of files copied to the ith cache disk from data disks in phase l.

With respect to the ith data disk, the utilization $\rho'_{il-data}$ in phase l is the sum of utilization caused by accessing files on the data disk and reading files to be duplicated to cache disks. Thus, $\rho'_{il-data}$ can be written as:

$$\rho'_{il-data} = \sum_{j=1}^{m'_i} \rho'_{ijl} + \frac{\sum_{j \in F_{il}^{M\_out}} t_{ijl}}{T} \tag{3.9}$$

where the first and second items on the right-hand side of Eq. 3.9 are the utilizations of accessing files and reading files from the data disk to make replicas on cache disks, respectively.

When it comes to the ith cache disk, the utilization $\rho'_{il-cache}$ in phase l is the sum of utilization contributed by accessed files and written file replicas to cache disks.

Thus, $\rho'_{il-data}$ can be written as:

$$\rho'_{il-cache} = \sum_{j=1}^{m'_i} \rho'_{ijl} + \frac{\sum_{j \in F_{il}^{M\_in}} t_{ijl}}{T} \tag{3.10}$$

where the first and second items on the right-hand side of Eq. 3.10 are the utilizations of accessing files and writing files to the cache disk to make replicas, respectively.

Modeling Power-State Transition Frequency for MAID

Eq. 4.1 in Section 3.2.5 shows that the power-state transition frequency adder is an important factor to model disk annual failure rate. The number of power-state transitions largely depends on I/O workload conditions in addition to the behaviors of MAID. In this subsection, we derive the number of power-state transitions from file access patterns.

We define the T_{BE} as the disk break-even time - the minimum idle time required to compensate the cost of entering the disk standby mode (T_{BE} values are usually anywhere between 10 to 15 seconds). Given file access patterns of the ith phase for a disk, we need to calculate the number τ_i of idle periods that are larger than the break-even time T_{BE}. The number of power-state transitions during phase i is $2\tau_i$, because there is a spin-down at the beginning of each large idle time and a spin-up by the end of the idle time. For an access pattern with N I/O phases, the total number of power-state transitions τ can be expressed as: $\tau = 2 \times \sum_{i=1}^{N} \tau_i$.

We model a workload condition where I/O burstiness can be leveraged by the dynamic power management policy to turn idle disks into the standby mode to save energy. To model I/O burstiness, we assume the first I/O requests of files within an access phase are arriving in a short period of time, within which disks are too busy to be switched into standby. After the period of high I/O load, there is an increasing number of opportunities to place disks into the standby mode. This workload model

allows MAID to achieve high energy efficiency at the cost of disk reliability, because the workload model leads to a large number of power-state transitions.

To conduct a stress test on reliability of MAID, we assume that the first requests of files on a disk arrive at the same time. For the first few time units, the workloads are so high that no data disks can be turned into standby. As the I/O load is descreasing, some data disks may be switched to standby when idle time intervals are larger than T_{BE}. In this workload model, MAID can achieve the best energy efficiency with the worst reliability in terms of the number of power-state transitions.

3.3.2 PDC- Popular Disk Concentration

Background

The PDC (Popular Data Concentration) technique proposed by Pinheiro and Bianchini migrates frequently accessed data to a subset of disks in a disk array [58]. Fig. 3.9 demonstrates the basic idea behind PDC: the most popular files are stored in the far left disk, while the least popular files are stored in the far right disk. PDC can rely on file popularity and migration to conserve energy in disk arrays, because several network servers exhibit I/O loads with highly skewed data access patterns. The migrations of popular files to a subset of disks can skew disk I/O load towards this subset, offering other disk more opportunities to be switched to standby to conserve energy. To void performance degradation of disks storing popular data, PDC aims to migrate data onto a disk until its load is approaching the maximum bandwidth.

The main difference between MAID and PDC is that MAID makes data replicas on cache disks, whereas PDC lays data out across disk arrays without generating any replicas. If one of the cache disks fails in MAID, files residing in the failed cache disks can be found in the corresponding data disks. In contrast, any failed disk in PDC can inevitably lead to data loss. Although PDC tends to have lower reliability than

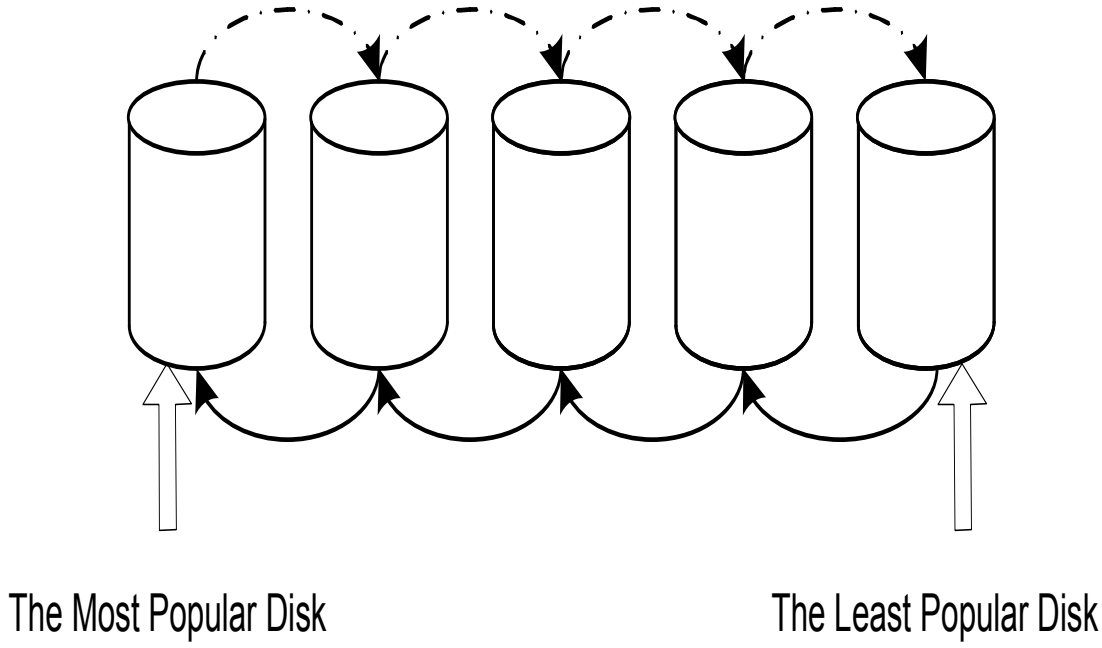

Figure 3.9: PDC System Structure

MAID, PDC does not need to trade disk capacity for improved energy efficiency and I/O performance.

Modeling Utilization of Disks in PDC

Since frequently accessed files are periodically migrated to a subset of disks in a disk array, we have to take into account disk utilization incurred by file migrations. Hence, the ith disk's utilization ρ'_{il} during phase l is computed as the sum of the utilization contributed by accessing files residing in disk i and the utilization introduced by migrating files to/from disk i. Thus, we can express utilization ρ'_{il} as:

$$\rho'_{il} = \sum_{j=1}^{m'_i} \rho'_{ijl} + \frac{Migration\ time}{T} \tag{3.11}$$

where T is the time interval of I/O phase l. The first and second items on the right-hand side of Eq. 3.11 are the utilizations caused by accessing files and duplicating files from data disks to cache disks, respectively.

To quantify utilization introduced by the file migration process (see the second item on the bottom-line on the right-hand side of Eq. 3.11), we define two set of files for the ith disk in the lth I/O phase. The first set $F_{il}^{M\_out}$ contains all the files migrated from disk i to other disks during the lth phase. Similarly, the second set $F_{il}^{M\_in}$ consists of files migrated from other disks to disk i in phase l.

Now we can formally express the utilization of disk i in phase l using the two file sets $F_{il}^{M\_out}$ and $F_{il}^{M\_in}$. Thus,

$$\rho'_{il} = \sum_{j=1}^{m'_i} \rho'_{ijl} + \frac{\sum\limits_{j \in F_{il}^{M}} t_{ijl}}{T_l} \tag{3.12}$$

where the second item on the right-hand side of Eq. 3.12 is the utilization incurred by (1) migrating files in set $F_{il}^{M\_out}$ from disk i to other disks and (2) migrating files in set $F_{il}^{M\_in}$ from other disks to disk i during phase l.

Modeling Power-State Transition Frequency for PDC

We used the same way described in Section 3.3.1 to model power-state transition frequency for PDC. Unlike MAID, PDC allows each disk to receive migrated data from other disks. In light of PDC, disks storing the most popular files are most likely to be kept in the active mode.

3.3.3 Results Evaluation

Experimental Setup

We developed a simulator to validate the reliability models for MAID and PDC. It might be unfair to compare the reliability of MAID and PDC using the same

number of disks, since MAID trades extra cache disks for high energy efficiency. To make fair comparison, we considered two system configurations for MAID. The first configuration referred to as MAID-1 employs existing disks in a parallel disk system as cache disks to store frequently accessed data. Thus, the first configuration of MAID improves energy efficiency of the parallel disk system at the cost of capacity. In contrast, the second configuration— called MAID-2—needs extra disks to be added to the disk system to serve as cache disks.

Our experiments were started by evaluating the reliability of PDC as well as MAID-1 and MAID-2. Then, we studied the reliability impacts of the proposed disk-swapping strategies on both PDC and MAID. We simulated PDC, MAID-1, and MAID-2 along with the disk-swapping strategies in two parallel disk systems described in Table 6.1. For the MAID-1 configuration, there are 5 cache disks and 15 data disks. In the disk system for the MAID-2 configuration, there are 5 cache disks and 20 data disks. As for the case of PDC, we fixed the number of disks to 20. Thus, we studied MAID-2 and PDC using a parallel disk system with 20 disks; we used a similar disk system with totally 25 disks to investigate MAID-1. We varied the file access rate in the range between 0 to 10^6 times per month. The average file size considered in our experiments is 300KB. The base operating temperature is set to 35°C. In this study, we focused on read-only workload. Nevertheless, the MINT model should be readily extended to capture the characteristics of read/write workloads.

Table 3.1: The characteristics of the simulated parallel disk system used to evaluate the reliability of PDC, MAID-1, and MAID-2.

Energy-efficiency Scheme	Number of Disks	File Access Rate (No. per month)	File Size (KB)
PDC	20 data (20 in total)	0˜10^6	300
MAID-1	15 data+5 cache (20 in total)	0˜10^6	300
MAID-2	20 data+5 cache (25 in total)	0˜10^6	300

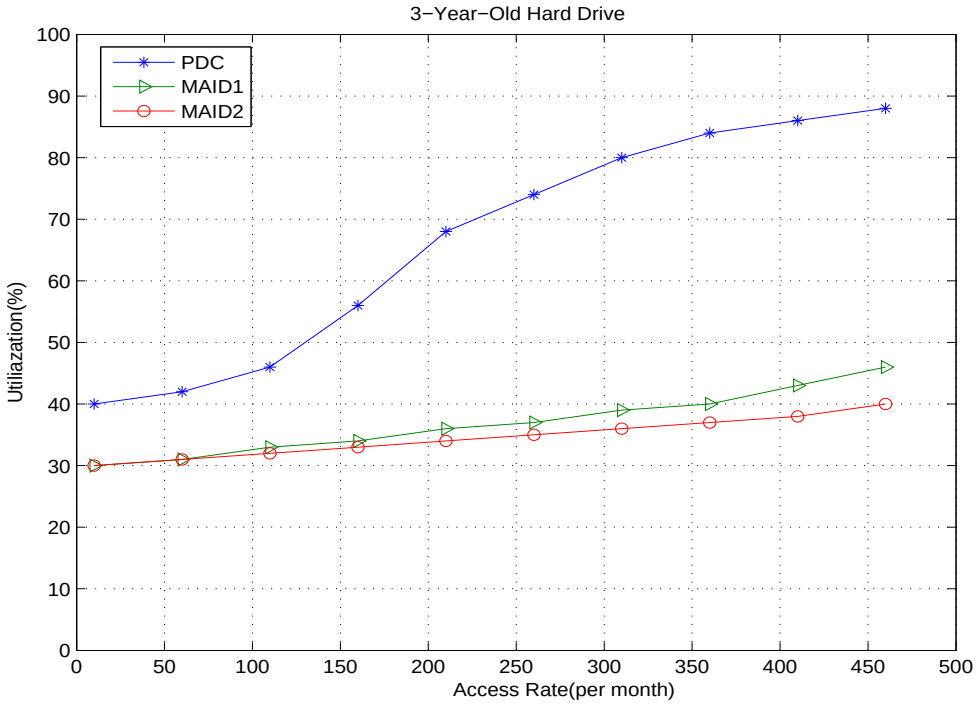

Figure 3.10: Utilization Comparison of the PDC and MAID
Access Rate(up to 500/month) Impacts on Utilization

Preliminary Results

In terms of utilization, when the file access rate of the files increase, represented in Fig. 3.10, the utilization of both PDC and MAID increase also. However, other than increasing as smoothly as that of MAID reaching nearly 50%, the utilization of PDC increases sharply hitting nearly 90%. The main reason is that PDC will be busy with migrating data in and out of the disks according to the popularity of the data. When the file access rate increases, which leads to more files migrating upward to the more popular disks while others migrating downward to the lease popular disks, the PDC system needs to spend more utilization to deal with the inner data migration in addition to the requests themselves. On the other hand, after copying the popular data to cache disks, there is no need for data disks to handle the requests in MAID any more. The increase of the curve is mainly influence by the utilization of cache disks in MAID. As one step further, Fig. 3.11 shows the annual failure rates

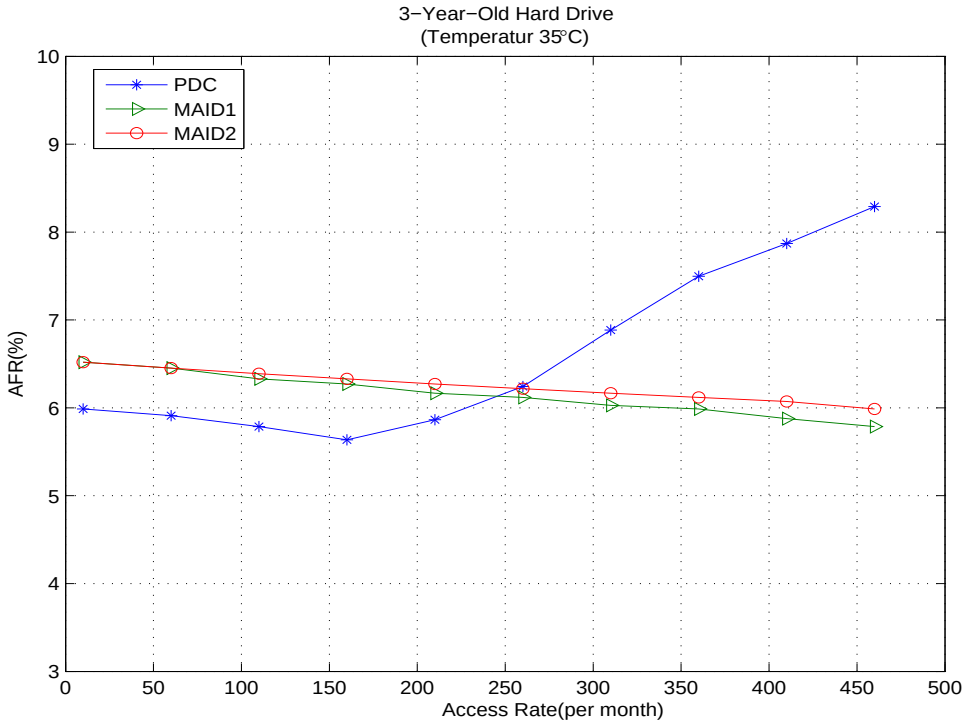

Figure 3.11: AFR Comparison of the PDC and MAID
Access Rate Impacts on AFR(Temperature=35°C)

of MAID1, MAID2, and PDC. We observe from Fig. 3.11 that the AFR value of PDC keeps increasing from 5.6% to 8.3% when the file access rate is larger than 150. We attribute this trend to high disk utilization due to data migrations. More interestingly, if the file access rate is lower than 150, AFR of PDC slightly reduces from 5.9% to 5.6% when the access rate is increased from 5 to 150. This result can be explained by the nature of the utilization function that is concave rather than linear. The concave nature of the utilization function is consistent with the empirical results reported in [61]. When the file access rate 150, the disk utilization is approximately 50%, which is the turing point of the utilization function.

Unlike PDC, MAID's AFR continues to decrease from 6.3% to 5.8% with the increasing file access rate. This declining trend might be explained by two reasons. First, increasing the file access rates reduces the number of power-state transitions.

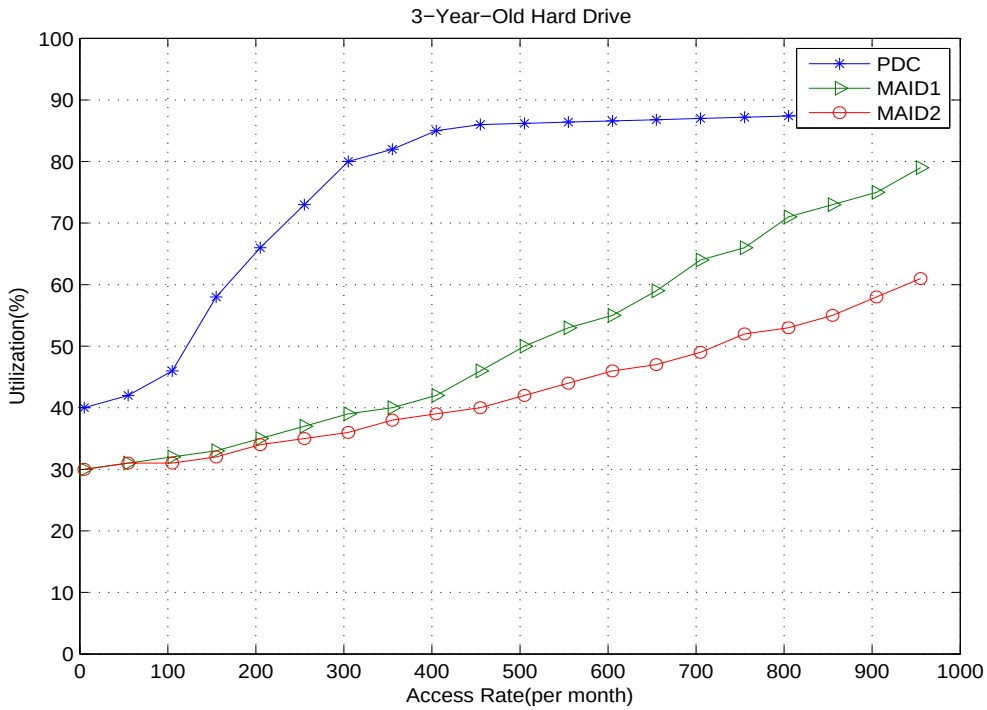

Figure 3.12: Utilization Comparison of the PDC and MAID
Access Rate(up to 1000/month) Impacts on Utilization

Second, the range of the disk utilization is close to 40%, which is in the declining part of the curve.

When the access rate is extended to 1000 per month, as shown in Fig. 3.12, the utilization of PDC gets close to 90% while those of MAID keep rising. The reason that utilization of MAID-1 grows faster than that of MAID-2 is because that when the method of weighted sum is adopted, the less number of disk is the more each disk weights more. As the systems utilization changed, the AFR will change accordingly. One important observation from Fig. 3.13 and Fig. 3.14 is that when access rate is higher than 700 times per month, the AFR of MAID-1 is getting higher than that of MAID-2. The reason is that the utiliaztion of MAID-1 keeps rising up over 60%, observed from Fig. 3.12, when access rate is higher than 700 times per month. And according to Fig. 3.3, the AFR will stop to rise up after utilization goes higher than 60%. Hence, we can predict that after access rate hit 900 per month, the AFR of

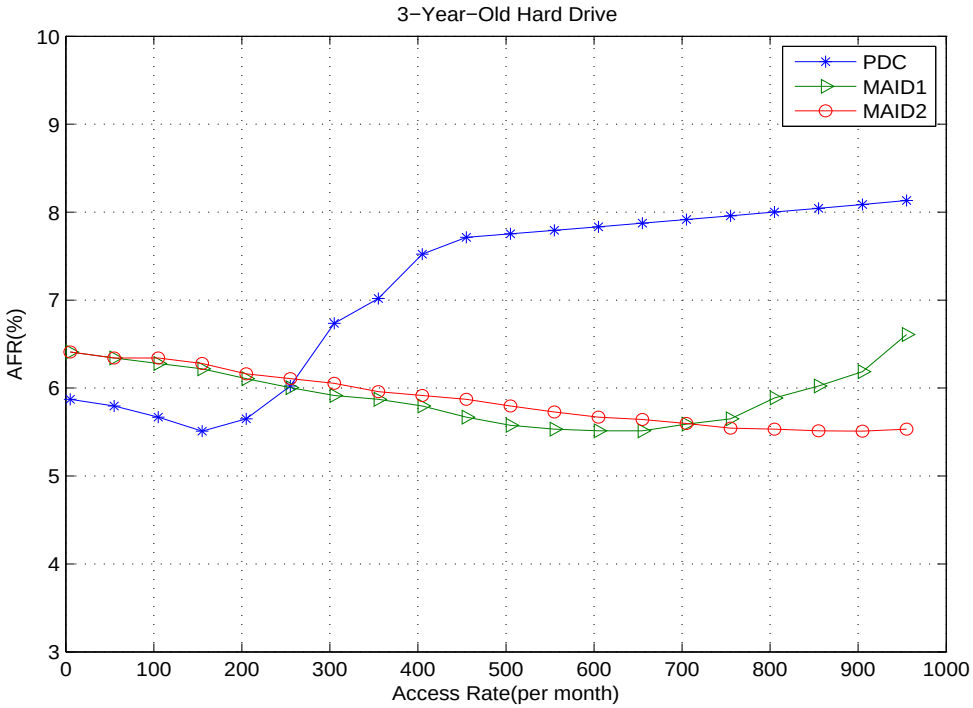

Figure 3.13: Utilization Comparison of the PDC and MAID Access Rate(up to 1000/month) Impacts on AFR(Temperature=35°C)

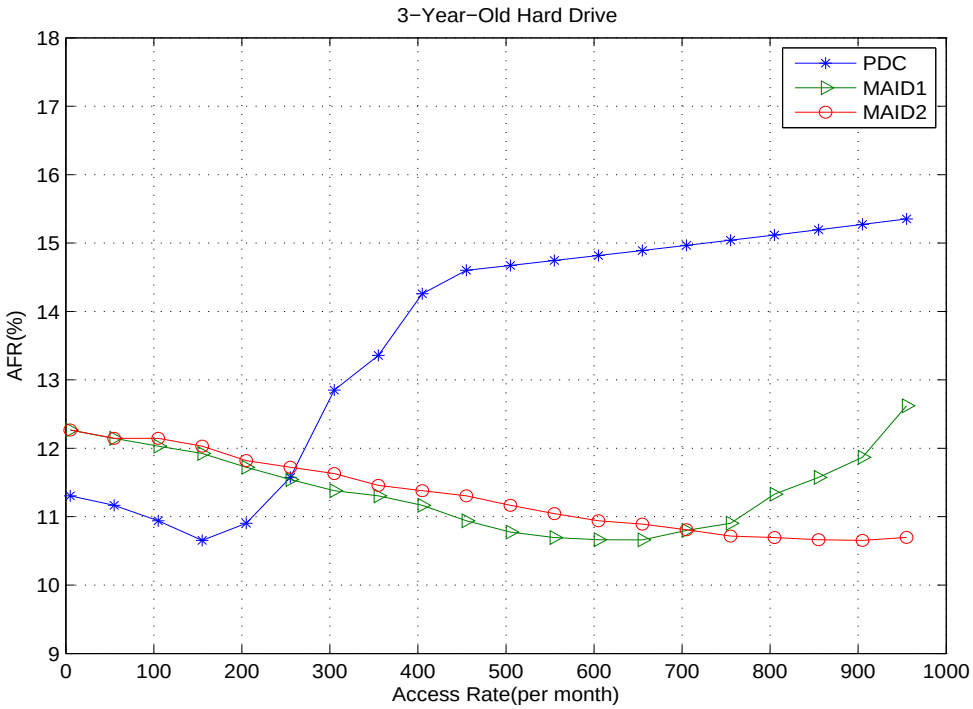

Figure 3.14: Utilization Comparison of the PDC and MAID Access Rate(up to 1000/month) Impacts on AFR(Temperature=40°C)

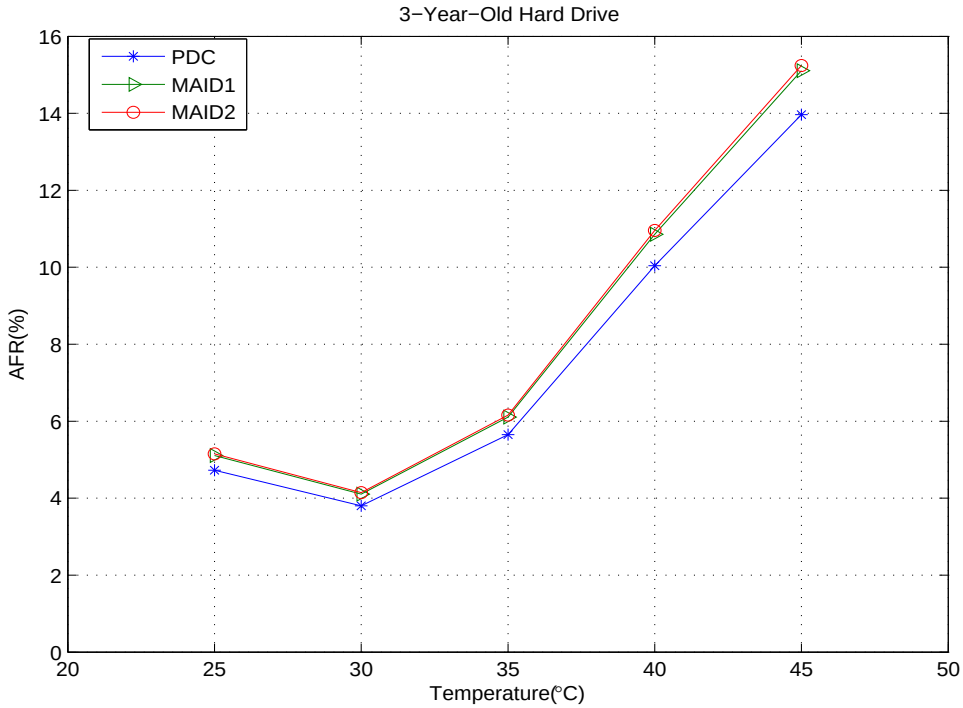

Figure 3.15: AFR Comparison of the PDC and MAID
Temperature Impacts on AFR (Access Rate= 200/month)

MAID-2 will be expected to stop to rise up. When we fix the access rate at 200 times per month and vary the temperature from 25°C to 45°C, as shown in Fig. 3.15, it is easy to see that as the temperature grows up, the AFR of all three systems goes down at the range of 25°C to 30°C, and goes up at the range of 30°C to 45°C. It is all according to the trend derived from Google [61]. Further, we notice that the AFR of PDC is lower than that of MAID. And when the temperature grows up, the AFR of MAID grows faster than that of PDC. On the contrary, when access rate is fixed at 450 times per month, as shown in Fig. 3.16, observation is that the AFR of PDC grows higher and faster than that of MAID. The two main reasons for these opposite results are utilization and frequency. As access rate is 200 times per month, even thought the utilization of PDC is higher than that of MAID, it still stays in the descending part of the utilization curve. From Fig. 3.3, it is obvious that higher utilization leads to lower AFR in the recession part of the curve. When access rate

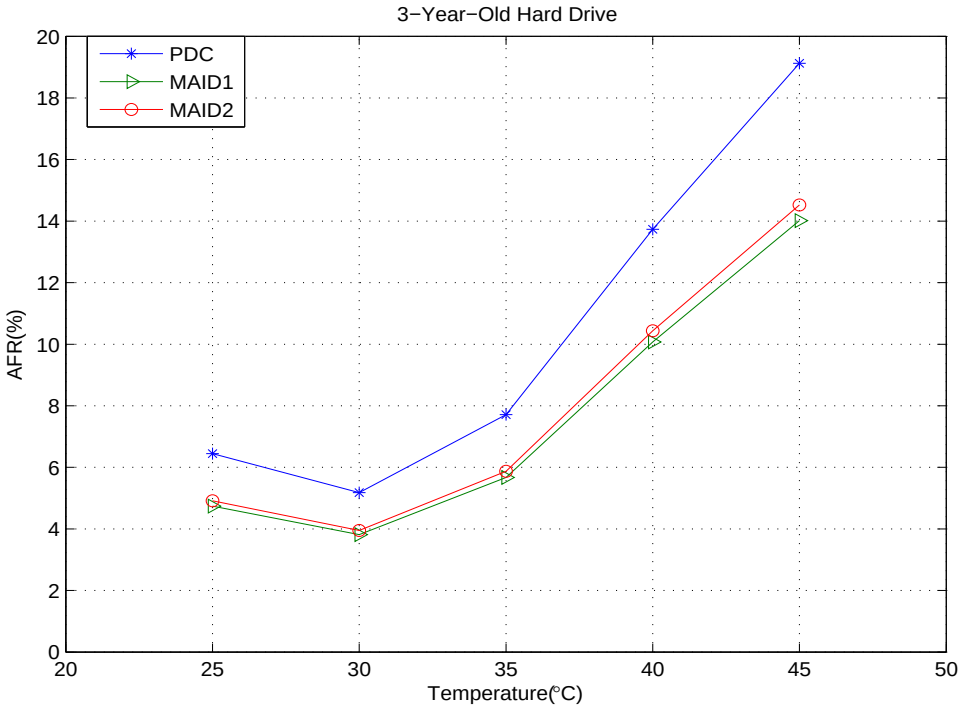

Figure 3.16: AFR Comparison of the PDC and MAID
Temperature Impacts on AFR (Access Rate= 450/month)

is 450 times per month, the utilization of PDC is approaching 90% because of the data migration, which is way higher than that of MAID as shown in Fig. 3.10. At this moment PDC stays in the ascending part of the utilization curve while MAID is about reaching the rock-bottom of the curve. Also, as the adder factor, the frequency makes the utilization of PDC grows even faster.

3.4 Summary

In recognition that existing disk reliability models cannot be used to evaluate reliability of energy-efficient disk systems, we propose a new model called MINT to evaluate the reliability of a disk array equipped with reliability-affecting energy conservation techniques. We first model the impacts of disk utilization and power-state transition frequency on reliability of each disk in a disk array. We then derive

the reliability of an individual disk from its utilization, age, temperature, and power-state transition frequency. Finally, we use MINT to study the reliability of disk arrays coupled with the MAID (Massive Array of Idle Disks) technique and the PDC (Popular Disk Concentration technique) technique.

Chapter 4

MREED: Reliability Analysis of An Energy-Aware RAID System

We develop a mathematical model– MREED– to quantitatively evaluate the failure rate of energy-efficient parallel storage systems. The Power-Aware Redundant Array of Inexpensive Disk (PARAID) aims to reduce energy use of commodity server-class disks without specialized hardware. The goal of PARAID is to skewed striping pattern to adapt to the system load by changing the number of powered disks. By spinning down disks during light workloads, PARAID can reduce power consumption, while still meeting performance demands. We show that MREED can be used to estimate a five-disk PARAID-0 system. We validate the accuracy of MREED using the DiskSim simulator. Our approach shows that MREED can rely on file access pattern to estimate system utilization correctly. Furthermore, even thought PARAID may achieve reasonable reliability, our model shows that PARAID's reliability is affected by data locality.

4.1 Motivations

Existing reliability models for conventional parallel and distributed disk systems do not consider energy-saving issues or data-stripping mechanisms. In this paper, we first study the reliability of a parallel disk system equipped with the PARAID [85] technique by employing the **M**athematical **R**eliability model for **E**nergy- **E**fficient RAI**D** system called MREED. As a mathematical model, MREED shows its advantage of presenting the reliability trend of energy-aware storage systems. However, it is challenging to validate the MREED model. To address the correctness issue of MREED, we validate the access-rate-utilization model, which converts file access rate

to utilization of the storage system, in MREED. Finally, we study impacts of the I/O load skewing technique –gear shifting – on the reliability of PARAID, a well known energy-aware data stripping storage system.

Existing energy conservation techniques can yield significant energy savings in disks. While several energy conservation schemes like cache-based energy-saving approaches normally have marginal impact on disk reliability, many energy-saving schemes (e.g., dynamic power management and workload skew techniques) inevitably have noticeable adverse impacts on storage systems [12][90]. For example, dynamic power management (DPM) techniques save energy by using frequent disk spin-downs and spin-ups, which in turn can shorten disk lifetime [22][34][46], redundancy techniques [60][102][82][89], workload skew [54][38][98], and multi-speed settings [32][76]. We pay attention on the reliability issue of RAID systems, existing energy conservation techniques can not be applied for RAID systems for the following reasons:

- Conventional RAIDs balance I/O load across all disks in the array for maximized disk parallelisms and performance, meaning that all disks are spinning even under a light load. No opportunity is offered to spin down any of disks;
- Server class disks are not designed for frequent power cycles, which significantly reduce life expectancy;
- Server systems cannot rely on caching and dynamic power management because the servers are too busy to have long idle time.

In this paper, our contributions are summaries as follows:

1. We propose a reliability model MREED for Power-Aware RAID (i.e., an energy aware data-stripping parallel storage system);
2. We introduce Weibull distribution analysis to MREED. Using the utilization of a storage system as an input, we can estimate and forecast the annual failure rate (a.k.a, AFR) of this system;

3. We validate the access-rate-utilization model of MREED;

4. We study the impacts of the gear-shifting schemes on the reliability of PARAID.

We study impacts of the I/O load skewing technique especially on PARAID-0, which is an energy-aware RAID-0 system. Experimental results shows that gear-shifting affects reliability of parallel disks due to two reasons: First, disks working at all gears tend to have high I/O utilization than disks that only works at high gears. Second, disks with high utilization are likely to have high risk of breaking down.

The remainder of this paper is organized as follows. Section 4.2 presents the overview of the MREED model. In Section 4.3, we apply MREED model to quantitatively estimate the reliability of PARAID. Section 4.4 presents experimental results and performance evaluation. Finally, Section 6.4 concludes the paper with discussions.

4.2 The MREED Modeling Framework

4.2.1 Overview

MREED is a framework developed to model reliability of parallel disk systems employing energy conservation techniques. In the MREED framework, we evaluate the reliability impacts of a specific energy-saving technique - the Power-Aware RAID. One critical module in MREED is to model the impact of energy-efficient schemes on the utilization and power-state transition frequency of each disk in a parallel disk system. Another important module developed in MREED is to calculate the annual failure rate of each disk as a function of the disk's utilization, power-state transition frequency. Given the annual failure rate of each disk in the parallel disk system, MREED is able to derive the reliability of an energy-efficient parallel disk system. As such, we used MREED to study the reliability of a parallel disk system equipped with the PARAID technique.

Fig. 4.1 outlines the MREED reliability modeling framework. MREED is composed of a Weibull-based disk reliability model, a system-level reliability model, and three reliability-affecting factors—temperature, power state transition frequency (hereinafter referred to as transition frequency or frequency) and utilization. Many energy-saving schemes inherently affect reliability-related factors like disk utilization and transition frequency. Given an energy optimization mechanism (e.g., PARAID [85]), MREED first converts data access patterns into the two reliability-affecting factors—frequency and utilization. The Weibull-based disk reliability model can derive individual disk's possibility of failure from utilization and power-on hours per year because these parameters are key reliability-affecting factors. Each disk's reliability is used as input to the system-level reliability model that evaluates the annual failure rate of parallel disk systems.

For simplicity without losing generality, we considered in MREED three reliability-related factors, namely: disk utilization, temperature, and power-state transitions. This assumption does not necessarily indicate by any means that there are only three parameters affecting disk reliability. Other factors having impacts on reliability include: handling, humidity, voltage variation, vintage, duty cycle, and altitude [25]. If a new factor has to be taken into account, one can extend the single reliability model by integrating the new factor with other reliability-affecting factors in MREED. Since the infant mortality phenomenon is out the scope of this study, we pay attention to disks that are no less than one year old.

The single-disk reliability can not be accurately described by one valued parameter because the disk drive reliability is affected by multiple factors. There are three major factors that affect disk reliability.

1. Disk Utilization can be characterized as the fraction of active time of a disk drive out of its total powered-on-time. The baseline value (i.e. R_{Base_Value} in Eq. 4.1) of AFR for a disk, which is derived from the Weibull distribution analysis, can

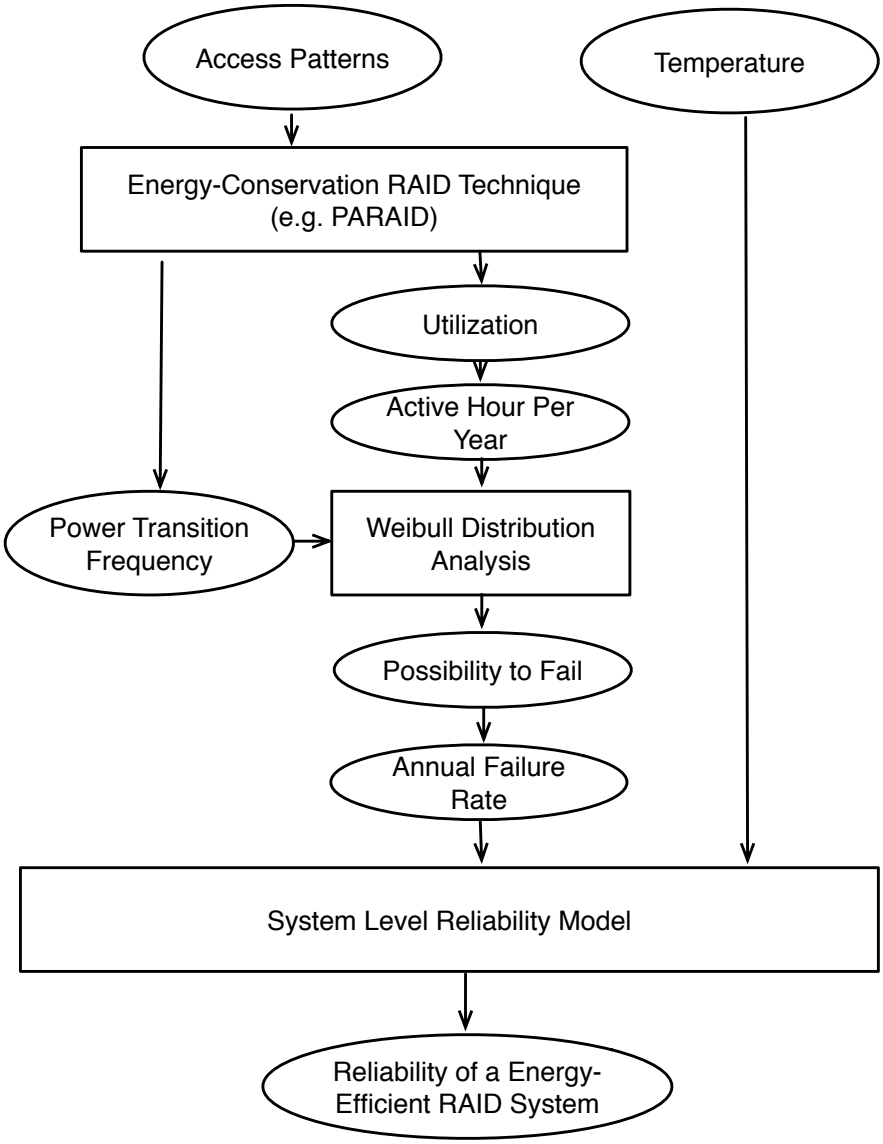

Figure 4.1: Overview of the MREED reliability modeling methodology

be calculated from the disk's utilization. The details will be discussed in the subsection 4.2.2;

2. Temperature, which acts as a multiplier to base failure rates in the MREED model. The temperature factor shown in the Table 4.1 was reported by Seagate Storage Group in Longmont, Colorado [20]. From the Table 4.1, we observe that as the temperature rises, the derating factor and the MTBF show clear

decreasing. In our research, we will use the Derating Factor(DF) as the Temperature Factor(i.e. $TemperatureFacotr$ in Eq. 4.1) of AFR. For example, at 30°C, the DF value is 0.78, which indicates that the AFR at this temperature is 22% higher than the AFR at 25°C. The main reason that we only use partial

Table 4.1: Temperature Factor

Temperature (°C)	Acceleration Factor	Derating Factor	Adjusted MTBF
25	1.0000	1.00	232,140
26	1.0507	0.95	220,553
30	1.2763	0.78	181,069
34	1.5425	0.65	150,891
38	1.8552	0.54	125,356
42	2.2208	0.45	104,463
46	2.6465	0.38	88,123

data from the report (25°C $\sim$ 46°C) is that we believe the cooling systems will prevent the temperature keeping higher than 46°C for long.

3. Power-State Transition Frequency, which is measured as the number of power-state transition (i.e. from active to standby or vice versa) per month. The reliability of an individual disk is affected by power-state transitions and, therefore, the increase in failure rate as a function of power-state transition frequency has to be added to a baseline failure rate (see Eq. 4.1 in the next subsection).

Hence, the failure rate R of an individual disk can be expressed as:

$$R = R_{Base\_Value} * \tau + \alpha * R_{Frequency\_Adder} \tag{4.1}$$

where R_{Base_Value} is the baseline failure rate derived from disk utilization, τ is the temperature factor, α is a coefficient to reliability R, and $R_{Frequency\_Adder}$ is the power-state transition frequency adder to the baseline failure rate, which can be calculated

by Eq. 4.2 [99].

$$R(f) = 1.51e^{-6}f^2 - 1.09e^{-5}f + 1.39e^{-2}, f \in [0, 500] \tag{4.2}$$

where f is a power-state transition frequency, $R(f)$ represents an adder to the base AFR value. For example, suppose the transition frequency is 300 per month, the base AFR value needs to be increase by 1.33%.

4.2.2 Weibull Distribution Analysis

Weibull distribution analysis is a leading method in the world for fitting life date. The primary advantage of Weibull analysis is the ability to provide accurate failure analysis and failure forecasts with extremely small samples [10]. It is now widely used reliability engineering and failure analysis including mechanical, electronic, materials, and human failures [21]. The Weibull reliability function describes the probability of survival as a function of time, and is described as follows in Eq. 4.3:

$$\begin{aligned} R(t) &= \int_t^\infty \frac{\beta(x)^{(\beta-1)}}{\theta^\beta} \exp[-(\frac{x}{\theta})^\beta]dx \\ &= \exp[-(\frac{t-\beta}{\theta})^\beta] \end{aligned} \tag{4.3}$$

where β is the shape parameter or slope parameter ($0 < \beta < \infty$), and θ is the scale parameter or characteristic life ($0 < \theta < \infty$). Given a disk drive's total power-on hours per year, and the utilization calculated by Eq. 4.1, we can calculate its total active hours during one year by Eq. 4.4

$$T_{active} = T_{power\_on} * \rho \tag{4.4}$$

where ρ is a disk utilization. With active hours as an input along with β and θ, we can use Eq. 4.3 to estimate its annual failure rate and MTBF (which serves as $BaseValue$ in Eq. 4.1).

4.3 Reliability Model for PARAID

4.3.1 Background

Different from traditional disk array systems, RAID balances the load across all disks in the array for maximized disk parallelism and performance [56]. In a RAID system, all disks are spinning even under a light load. Instead of spinning down inactive disks under a light load as MAID [19] or PDC [58] behave, PARAID exploits unused storage to replicate and stripe data blocks in a skewed fashion, so that disks can be organized into hierarchical overlapping sets of RAIDs. Each set contains a different number of disks, and can serve all requests via either its data blocks or replicated blocks. PARAID introduces a skewed striping pattern that allows RAID devices to use just enough disks to meet the system load. Each set is analogous to a gear in automobiles as PARAID has aggregated disk bandwidth. PARAID varies the number of powered-on disks via *gear-shifting* among sets of disks to reduce power consumption [85]. The authors confirmed that PARAID system can save up to 34% energy compared to the conventional 5-disk RAID system. However, such energy-efficient technique may have adverse impacts on the reliability of the storage system. The system has to spend extra disks utilization on copying data from disks that are about to be spun down, which leads to higher risk of system failures. Furthermore, after a gear-shifting down, less number of disks will provide the same amount of service as it is before the gear-shifting, which pushes the power-on disks into higher utilization range and thus makes the system even less reliable. Thirdly, due to the data stripping technique, each single disk in the PARAID system only holds part of files. PARAID may face absolute data lose if the number of failure disks exceeds the

system's failure tolerance. The reliability issue of PARAID counts much more than conventional disk array systems. Fig. 4.2 is a PARAID system consists of four disks.

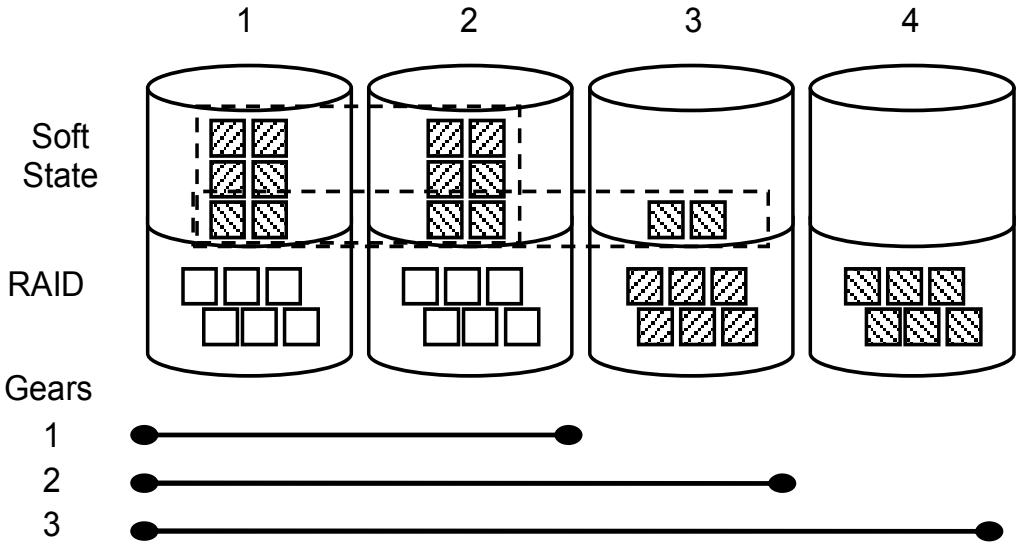

Figure 4.2: Framework of PARAID: skewed striping of replicated blocks in soft state, creating 3 RAID gears over 4 disks
[85]

Fig. 4.2 shows that each disk in PARAID has two separate states– the Soft State and RAID State. When operating in gear 3, with all four disks powered, PARAID works as the way of conventional RAID system offering maximized disk parallelism and performance accordingly. As I/O load decreases, PARAID down-shifts into gear 2 by spinning down the fourth disk. Before the down-shifting, the blocks stored in the RAID states on disk 4 are copied to disk 1$\sim$3 one by one. In this case, disk 1 holds the $1st$ and the $4th$ block of disk 4, disk 2 keeps the $2nd$ and the $5th$ block of disk 4, and disk 3 will store the $3rd$ and the $6th$ block of disk 4. If the load keeps decreasing, PARAID will further down-shift into gear 1 by powering down the third disk.

4.3.2 Modeling Utilization of Disks in PARAID

Recall that the annual failure rate of each disk can be calculated using utilization, operating temperature as well as power-state transition frequency. To model reliability of a disk array equipped with PARAID, we have to first address the issue of modeling disk utilization used to calculate base annual failure rates (R_{Base_Value} in Eq.4.1 shown in Section 4.2). In this subsection, we develop a utilization model capturing behaviors of a RAID-based disk array. The utilization model takes file access patterns as an input and calculates the utilization of each disk in the disk array.

Disk utilization is computed as the fraction of active time of a disk drive out of its total powered-on-time. Now we describe a generic way of modeling the utilization of a disk drive. Let us consider a sequence of I/O accesses with L I/O phases. We denote T_l as the length or duration of the lth I/O phase. Without loss of generality, we assume that a file access pattern in an I/O phase remains unchanged. The file access pattern, however, may vary in different phases. The relative length or weight of the ith phase is expressed as $W_l = T_l/T$ where $T = \sum_{l=1}^{L} T_l$ is the total length of all the I/O phases. Suppose the utilization of a disk in the lth phase is ρ_l, we can write the overall utilization ρ of the disk as the weighted sum of the utilization in all the I/O phases. Thus, we have

$$\rho = \sum_{l=1}^{L} (W_l \times \rho_l) = \sum_{l=1}^{L} (\frac{T_l}{T} \times \rho_l) \tag{4.5}$$

Since a PARAID system requires at least two disks to achieve the minimum I/O parallelism, the PARAID system consists of N disks has $(N-1)$ gears to shift. Assume that at the $G_{N-1}th$ gear, in which case all N disks of the system are kept spinning in order to offer the maximum parallelism, each single disk stores M blocks. When disk N is spun down, all its M blocks will be separated into $N-1$ sets in a way that each of the rest $N-1$ disks will handle making replicas for M blocks in

disk N. Thus, we have:

$$F_{out\_G_{(N-1)(N-2)}} = M \tag{4.6}$$

and

$$if \mod (\frac{F_{out\_G_{(N-1)(N-2)}}}{N-1} = 0)$$

$$F_{in\_G_{(N-1)(N-2)}} = \frac{F_{out\_G_{(N-1)(N-2)}}}{N-1}$$

$$else \begin{cases} F_{in\_G_{(N-1)(N-2)}} = \left\lfloor \frac{F_{out\_G_{(N-1)(N-2)}}}{N-1} \right\rfloor + 1 \\ \quad for\ disk\ 1 \sim disk\ D \\ F_{in\_G_{(N-1)(N-2)}} = \left\lfloor \frac{F_{out\_G_{(N-1)(N-2)}}}{N-1} \right\rfloor \\ \quad for\ rest\ of\ (N-D) disks \end{cases} \tag{4.7}$$

where $D = mod(\frac{F_{out\_G_{(N-2)(N-3)}}}{N-2})$, $F_{out\_G_{(N-1)(N-2)}}$ represents replicas of the blocks moved out from the disk N when PARAID shifts down the gear from G_{N-1} to G_{N-2} due to the decreasing workload. $F_{in_G_{(N-1)(N-2)}}$represents the set of replicated blocks that moved into each of the $N-1$ disks. If M can be exactly divided by $N-1$, each disk will handle $M/(N-1)$ blocks. Otherwise, the first remainder of $M/(N-1)$ disks will handle one extra block, while each of the rest disks will handle quotient of $M/(N-1)$ blocks.

Similarly, when PARAID shifts down from gear G_{N-2} to G_{N-3}, we have:

$$F_{out\_G_{(N-2)(N-3)}} = M + F_{in\_G_{(N-1)(N-2)}} \tag{4.8}$$

and

$$if \mod (\frac{F_{out\_G_{(N-2)(N-3)}}}{N-2} = 0)$$

$$F_{in\_G_{(N-2)(N-3)}} = \frac{F_{out\_G_{(N-2)(N-3)}}}{N-2}$$

$$else \begin{cases} F_{in\_G(N-2)(N-3)} = \left\lfloor \frac{F_{out\_G_{(N-2)(N-3)}}}{N-2} \right\rfloor + 1 \\ \qquad\qquad = \left\lfloor \frac{M + F_{out\_G_{(N-1)(N-2)}}}{N-2} \right\rfloor + 1 \\ \qquad for\ disk\ 1 \sim disk\ D \\ F_{in\_G_{(N-2)(N-3)}} = \left\lfloor \frac{F_{out\_G_{(N-2)(N-3)}}}{N-2} \right\rfloor \\ \qquad\qquad = \left\lfloor \frac{M + F_{out\_G_{(N-1)(N-2)}}}{N-2} \right\rfloor \\ \qquad for\ rest\ of\ (N-D) disks \end{cases} \tag{4.9}$$

It is noticed that the disk to be powered off needs to duplicate blocks, which were copied during the first downshifting period of time, apart from its own M blocks. The rest $N-2$ disks move in more replicated blocks accordingly.

In general, when PARAID shifts down from gear G_j to G_i, where $j \in (3, ..., N-2)$, the number of blocks that the disk to be powered off must handle the following number of reads copy out is

$$\begin{aligned} F_{out\_G_{(j)(j-1)}} =& M + F_{in\_G_{(N-1)(N-2)}} + F_{in\_G_{(N-2)(N-3)}} + \\ &+ F_{in\_G_{(N-3)(N-4)}} ... + F_{in\_G_{(j+1)(j)}} \end{aligned} \tag{4.10}$$

while the number of blocks that must be written to the rest $j-1$ disks is expressed as:

$$if \bmod (F_{out\_G_{(j)(j-1)}}/j) = 0$$

$$F_{in\_G_{(j)(j-1)}} = F_{out\_G_{(j)(j-1)}}/j$$

$$else \begin{cases} F_{in\_G_{(j)(j-1)}} = \left\lfloor F_{out\_G_{(j)(j-1)}}/j \right\rfloor + 1 \\ \quad for\ disk\ 1 \sim disk\ mod(F_{out\_G_{(j)(j-1)}}/(j-1)); \\ F_{in\_G_{(j)(j-1)}} = \left\lfloor F_{out\_G_{(j)(j-1)}}/j \right\rfloor \\ \quad for\ rest\ of\ disks. \end{cases} \tag{4.11}$$

where j represents the current gear number while $(j-1)$ indicated the gear number that the PARAID system is about to be shifted to, $\lfloor F_{out\_G_{(j)(j-1)}}/j \rfloor$ returns the integral part of $F_{out\_G_{(j)(j-1)}}$. We assume that every single file has the same number of blocks, each of which has the same size. Hence, the I/O time for accessing each single block is the same. Now we can formally express the utilization of disk i in phase l as follows:

For the disk to be power-off, we have:

$$\rho_{power-off} = \frac{T_{I/O} + T_{read}}{T} \tag{4.12}$$

, while for the rest of disks, we have:

$$\rho_{power-on} = \frac{T_{I/O} + T_{write}}{T} \tag{4.13}$$

To improve the readability, Table 4.2 lists the notation used in our model.

Table 4.2: List of Notations

Parameter	Description
R	Total Reliability
R_{Base_Value}	Reliability of Utilization
$R_{freq}(f)$	Reliability of Power Transition Frequency f
τ	Temperature Factor
α	Coefficient to R
β	Shape Parameter
θ	Scale Parameter
T_{active}	Active Time
T_{power_on}	Power-on Time
ρ	Disk Utilization
W_l	Relative Weight of l-th I/O phase
F_{out}	Copy Out File
F_{in}	Copy In File
N	Number of Disks
M	Number of Blocks
$T_{I/O}$	Service Time for I/O Requests
T_{read}	Service Time for Reading Duplicated Files
T_{write}	Service Time for Writing Duplicated Files

4.4 Reliability Evaluation

4.4.1 Experimental Setup

We developed a simulator in which the PARAID-0 system (a.k.a Power-Aware RAID Level 0) is implemented. Table 4.3 shows the parameters of configurations for PARAID-0. We evaluate the reliability of a five-disk PARAID-0 system, in which the highest gear of the system is 4. In order to keep the RAID-0 configuration, there are two disks kept active at the lowest gear 1. The file access rate is generated by Poisson distribution. The operating temperature is set to 38°C. Furthermore, we are using properties of Seagate hard disk drive in our simulator. The properties are also shown in Table 4.3. Since Seagate's disks properties are introduced to our experimental setups, we set $\beta = 0.55$, $\theta = 8410332$ in the Weibull analysis model, and 0.54 as the τ, which is the temperature factor in Eq. 4.1 [20].

Table 4.3: Experiment Parameter Setup

Disk Type	SEAGATE ST3146855FC
Capacity	146 GB
Cach Size	SATA 16MB
Buffer to Host Transfer Rate	4Gb/s) (MAX)
Total Number of Disks	5
File Size	100 MB
Number of Files	1000
Synthetic Trace	Poisson Distribution
Time Period	24 hours
Interval Time (Time Phase)	1 hour
Power On Hour Per Year	8760

4.4.2 Disk Utilization

We first investigate the impacts of file access rate (λ in Poisson distribution) on utilization of PARAID-0. We set values of utilization to trigger gear-shifting are set to 60% for gear up while 30% for gear down. The PARAID-0 is assumed to be started at the top gear– all five disks are working . Fig. 4.3 plots the utilization comparison of PARAID-0 and RAID-0 within 24 hours. The average access rate is set to 20 per hour ($\lambda = 20$), which is relatively low. We observe from Fig. 4.3 that as time goes, the utilization of RAID-0 stays stable around 22%, while that of PARAID-0 increases twice then stays stable around 36%. Those two increasing points are caused by the gear-down shiftings hence the decreasing the number of active disks. Even though the utilization of PARAID-0 is 60% higher than that of RAID-0 at the end hour 24, the energy consumption of PARAID-0 is 40% lower than that of RAID-0 since there are only three active disks by then. Fig. 4.4 shows the utilization comparison of PARAID-0 and RAID-0 when the average access rate is set to 80 per hour ($\lambda = 80$), which is 3 times higher than that in Fig. 4.3. From the figure we notice that the utilization difference between PARAID-0 and RAID-0 is very vague. The major reason is that when the access rate is relatively high enough, the utilization of PARAID-0 keeps high (around 90% shown in Fig. 4.4) accordingly and, therefore a gear-shifting mechanism

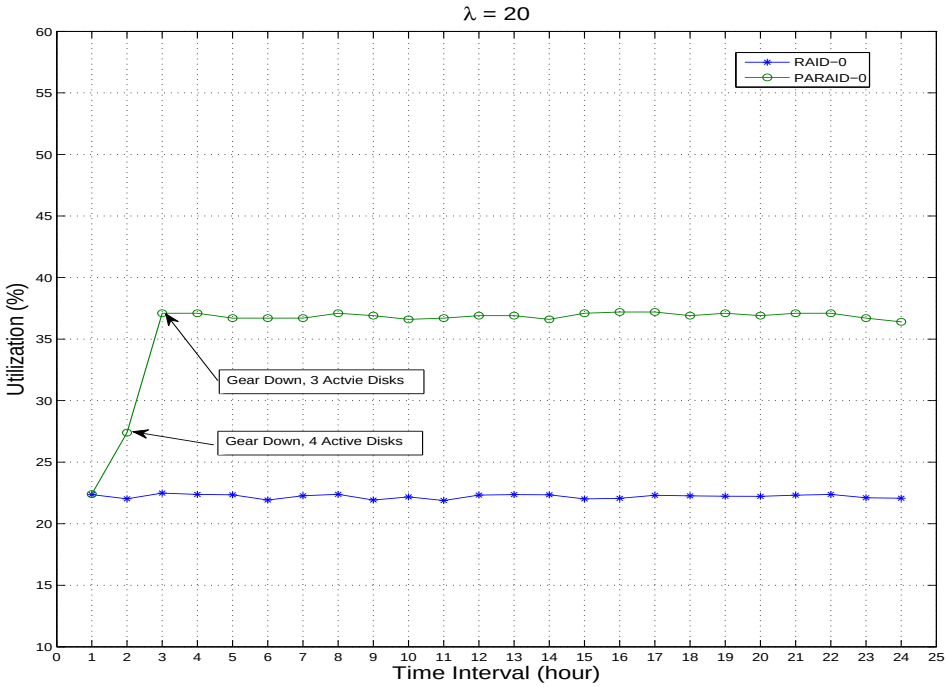

Figure 4.3: Disks Utilization Comparison Between PARAID-0 And RAID-0 at A Low Access Rate(20 times per hour)

is not triggered. Hence at high access rate pattern, PARAID-0 behaves as similar as the regular RAID-0 system.

4.4.3 Annual Failure Rate

Fig. 4.5 illustrates the annual failure rates (AFR) of PARAID-0 and RAID-0 based on their utilization which is derived from Fig. 4.3. Results plotted in Fig. 4.5 show that AFR values of RAID-0 keeps increasing from 4.5% to 5.46% when hour lapses, while AFR of PARAID-0 increases by 4% at hour 2 and surges by another 8% at hour 3. We attribute this trend to the decreasing of the number of active disks due to gear-down shifings. Since the utilization of PARAID-0 keeps the same as that of RAID-0 at high access rate, the AFR of the two systems are similar to each other accordingly. However, if the power transition issue is taken into account, AFR of PARAID-0 is different from that of RAID-0 even if their access rate are the same to

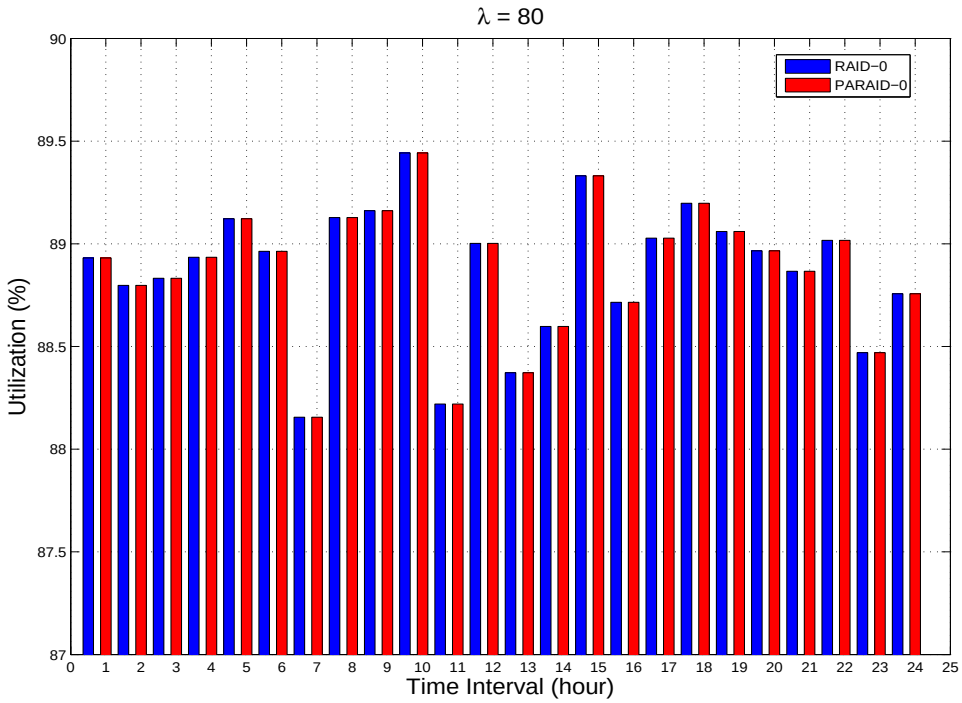

Figure 4.4: Disks Utilization Comparison Between PARAID-0 And RAID-0 at A Low Access Rate(80 times per hour)

each other. Fig. 4.6 reveals the AFR comparisons between RAID-0 and PARAID-0 starts from different gears within 24 hours. From the figure we observe that when the access rate increases shapely if PARAID-0 is not at the top gear, AFR of the system will suffer from the number of power transitions. Storage system at lower gear hvae relatively poor reliability. It is mainly because that more disks needs to be spun on to meet the needs of requests hence more number of power transitions will be counted.

4.5 Summary

This paper presents a reliability model called MREED to quantitatively study the reliability of energy-efficient parallel disk systems equipped with the PARAID technique. Note that PARAID is a newly developed energy-saving scheme for RAID systems. It aims to skew I/O load towards a few disks so that other disks can be transitioned to low power states to conserve energy. I/O load skewing techniques like

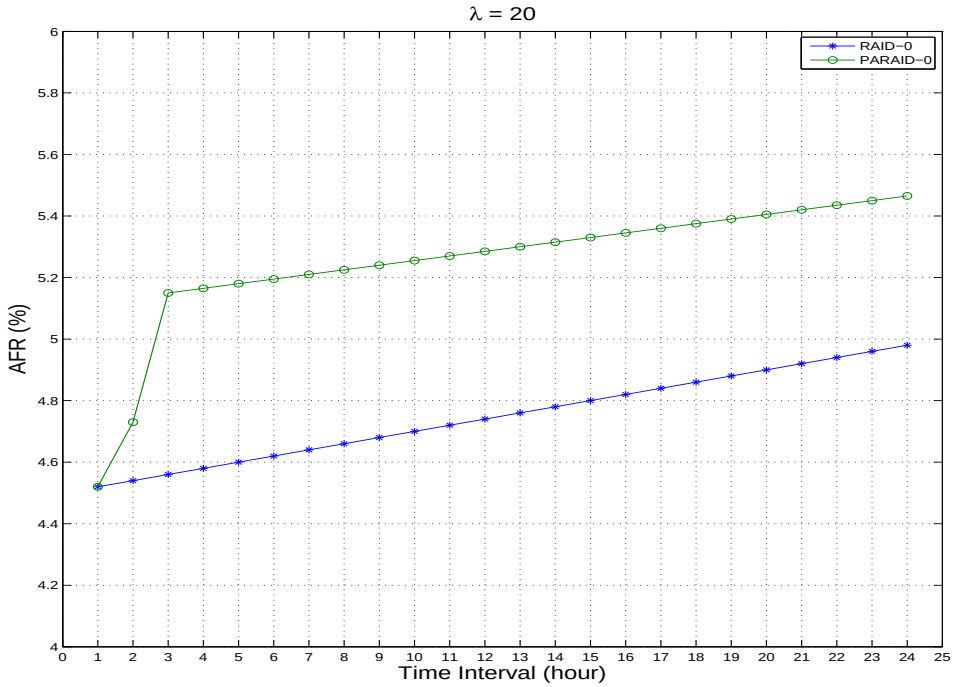

Figure 4.5: AFR Comparison Between PARAID-0 And RAID-0 at A Low Access Rate(20 times per hour)

PARAID inherently affect reliability of RAID disks, because disks keep working on low gears tend to have high failure rates, let alone the risk of failure caused by data duplicating during the *gear shifting*. Furthermore, once the number of failed disks exceeds the system's tolerance, data in the system are lost without any chance of being recovered. To address the model validation issue for MREED, we modified the DiskSim simulator, which is a widely-used storage system simulator, to validate our access-rate-utilizaiton sub-model of MREED by comparing the utilization of 5-disk PARAID system using a real-world disk I/O trace with the utilization that calculated from the MREED model using the same trace.

Future directions of this research can be performed in the following. First, we will extend the MREED model to investigate reliability of different levels (e.g., level 5) of PARAID in the future which introduces parity data technique to tolerate one disk failure. Second, we will investigate a fundamental trade-off between reliability

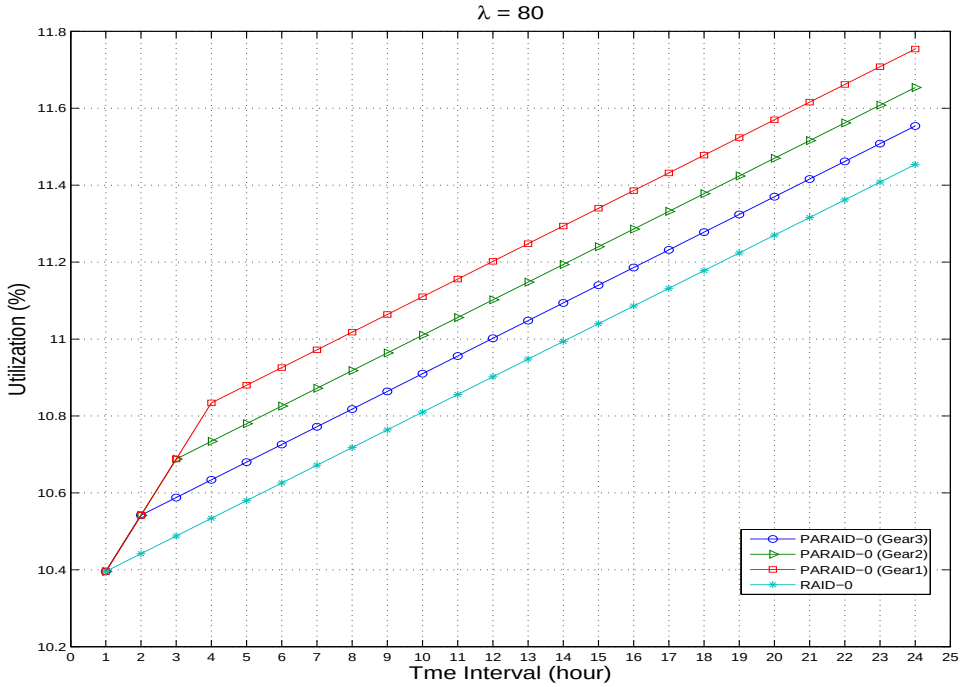

Figure 4.6: AFR Comparison Between PARAID-0 And RAID-0 at A High Access Rate(80 times per hour)

and energy-efficiency in the context of energy-efficient RAID systems. A tradeoff curve will be used as a unified framework to justify whether or not it is wise to trade reliability for high energy efficiency. Last, we will evaluate and compare an array of energy-saving techniques with respect to specific application domains.

Chapter 5

Models Validation

5.1 Model Validation

5.1.1 The Validation Techniques

It is reasonable to use MINT to compare the reliability performance of different energy-efficient storage systems, because the reliability models of the MAID and PDC storage systems use the same experimental data. It is challenging to validate the accuracy of the MINT modeling framework, since we are unable to watch MAID and PDC running for a couple of decades. One way to address this problem is to maintain and monitor a large number of MAID and PDC systems for a short period of time (e.g., 5 to 10 years). If one can watch the MAID and PDC systems over their entire service life, failure-rate data will be collected to validate reliability models. Even if we can test MAID and PDC with 100 disks for five years, the sample size is still considered small.

To address this validation problem, we verify MINT using the combination of the following two validation techniques [68], which are practical approaches to verification and validation of models.

- **Event Validity:** Events of occurrences of the model are compared to those of the real storage system to determine if they are similar. For example, in our validation process, we compared the file access rates in a real-world file system.
- **Historical Data Validation:** We first used part of the historical file access data (i.e., file I/O traces) for building our models. Then, we relied on the remaining data to test the models.

Recall that MINT consists of two major components - the utilization model (see Sections 3.3.1 and 3.3.2) and the failure-rate model. The utilization model estimates disk utilization of the MAID and PDC systems based on I/O access rates. The failure-rate model relies on real world failure data (see [61]) to predict the failure rate of a disk from its utilization.

To validate MINT, we have to validate the utilization model and the failure-rate model. Since failure rates in this study are projections based on the failure-rate model derived from Google's empirical analysis (see [61]), we pay attention on the validation of the utilization model.

We performed the following six steps repeatedly to validate the utilization model described in Sections 3.3.1 and 3.3.2.

- **Step 1:** We made use of the real-world I/O trace (i.e., Berkeley web trace) to derive file access rates.
- **Step 2:** The file access rates are applied to our utilization model to estimate disk utilizations of the MAID and PDC storage systems.
- **Step 3:** We implemented a trace replay tool, which captures the rapid evolution of web server workloads.
- **Step 4:** We developed the simple MAID and PDC systems that handle I/O requests created by the trace reply tool.
- **Step 5:** The utilizations of disks in the MAID and PDC storage systems are measured.
- **Step 6:** We compare the measured disk utilizations from the two real storage systems (see Step 5) with the disk utilizations derived from our models (see Step 2).

5.1.2 Berkeley Web Trace Replay

The Berkeley Web Trace [2] used in the model validation procedure was collected from a web server for an online library project from January 22nd to February 23rd, 1997. The Berkeley Web Trace data represents intensive I/O activities of a real-world system, for which MAID and PDC can conserve energy. Because I/O access rates in this study are measured in term of number I/O per/month or No./month, we decided to replay a one-month trace containing 33 trace files and 25205132 I/O requests. Among all the requests, 24481520 are file accesses requesting 302519 web files. The trace replay period is 1631753 seconds or 453.3 hours.

Table 5.1: File Access Rates of the One-Month Web Trace

File Access Rate Interval (No./Month)	The number of files
$0 \sim 10$	185383
$10 \sim 10^2$	112203
$10^2 \sim 10^3$	4539
$10^3 \sim 10^4$	244
$10^4 \sim 10^5$	113
$10^5 \sim 10^6$	33
$10^6 \sim 10^7$	4

Before applying file access rates into the utilization models presented in Sections 3.3.1 and 3.3.2, we performed an analysis on file access rates of the web traces. The goal of this analysis is to determine the access rate of each web file accessed over the one month period. Table 5.1 summarizes the distribution of file access rates of the 12304467 web files recorded in the 33 traces. Table 5.1 indicates that a vast majority (i.e., more than 61%) of web files were accessed less than ten times within a month. However, there are a few web files that were accessed for more than 1000 times over a one-month period. The analysis result shows that the highest file access rate is 3180697 No./month.

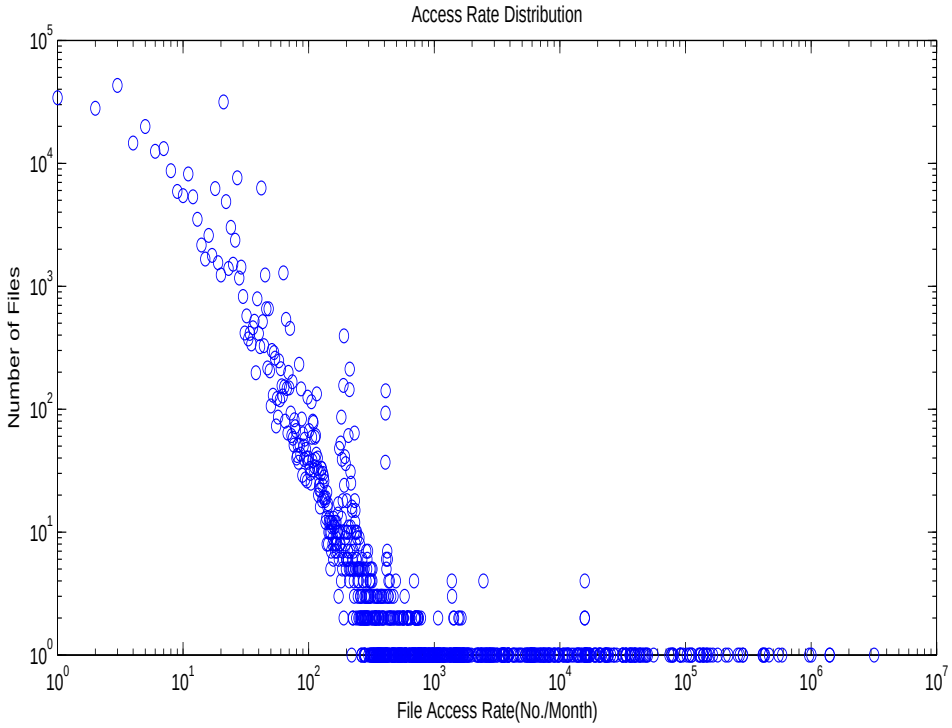

Figure 5.1: The file access rate distribution of the one-month Berkeley web trace. Access Rate ranges from 1 to $4.5 * 10^4$ No./month

Fig. 5.1 shows the files accesses distribution pattern using a bar chart. The distribution pattern suggests that when the access rate increases, the number of files that have such access rate decreases dramatically.

5.1.3 Experimental Results

Since the Utilization-AFR model, which transfers the utilization of systems to reliability, is employing the same date from the validated Google report, we only show the validation of Access Rate-Utilization model in this subsection.

Fig. 5.2 indicates the utilization comparison between the MINT model and Berkeley Web Trace-driven simulation. In order to make a clearer comparison between the MINT model and the trace-driven simulation, we divided the utilizaiton comparison of PDC, MAID-1 and MAID-2 separately (as shown in Fig. 5.3,Fig. 5.4 and Fig. 5.5). From the figures, we observed that the the cures according to MINT model show a

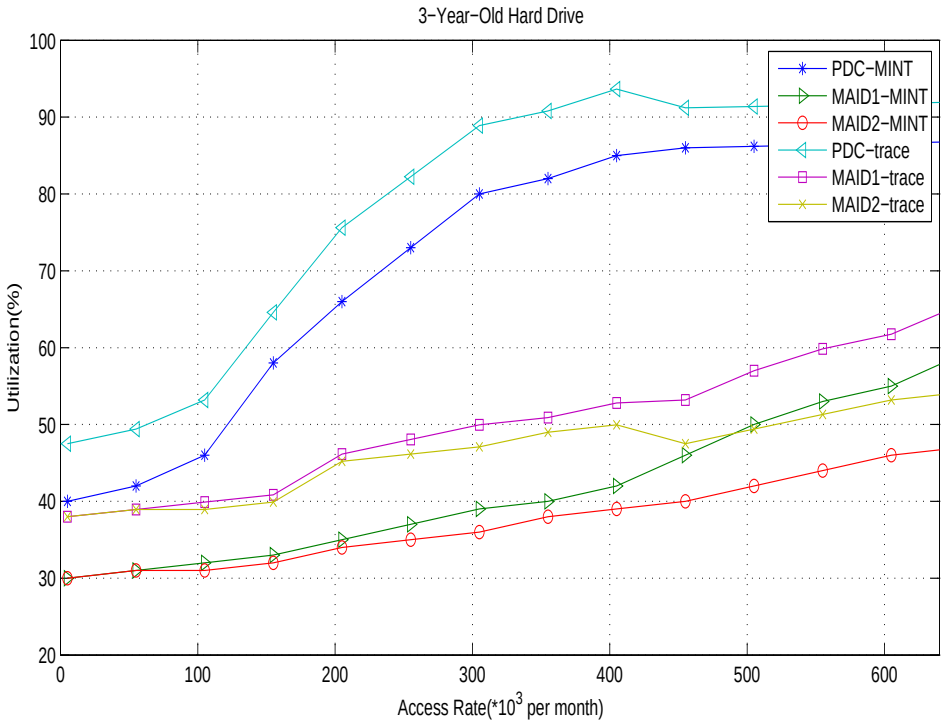

Figure 5.2: Impacts of file access rate on disk utilization. Access rate varies from 10 to $64 * 10^4$ No./month

similar trend to that of simulation. Furthermore, the differential rate between the model and the simulation is around 10%.

After validating the Access Rate-Utilization sub-model, we further present the comparison results of Access Rate-AFR between the MINT model and the simulation. We are able to build up a Utilization-AFR sub-model of our own and insert it to our MINT model. However, due to the lack of maintenance date recently, how to validate the sub-model becomes a hard issue to deal with. Instead, we are using the validated data published by Google [61] in this part. Once we get more updated data in the future, such sub-model could be re-modified.

Fig. 5.6, Fig. 5.7, and Fig. 5.8 show the impacts of file access rate on AFR. Even thought the trends of Access Rate-Utilization sub-model appeared similar between the model and the simulation (as shown in Fig. 5.3,Fig. 5.4 and Fig. 5.5), there

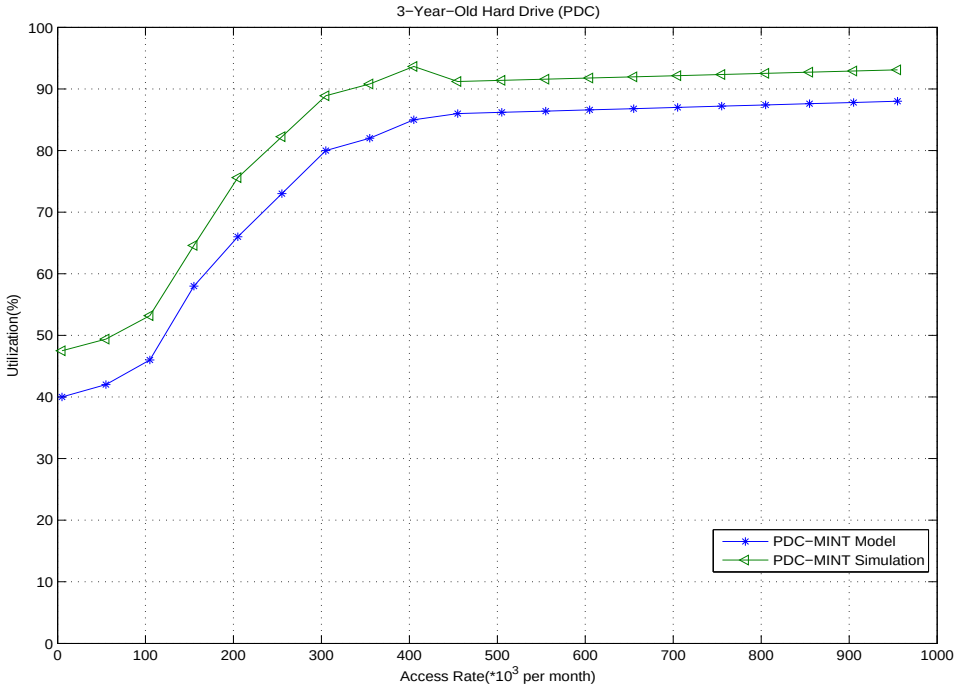

Figure 5.3: Impacts of file access rate on disk utilization (PDC). Access rate varies from 10 to $64 * 10^4$ No./month

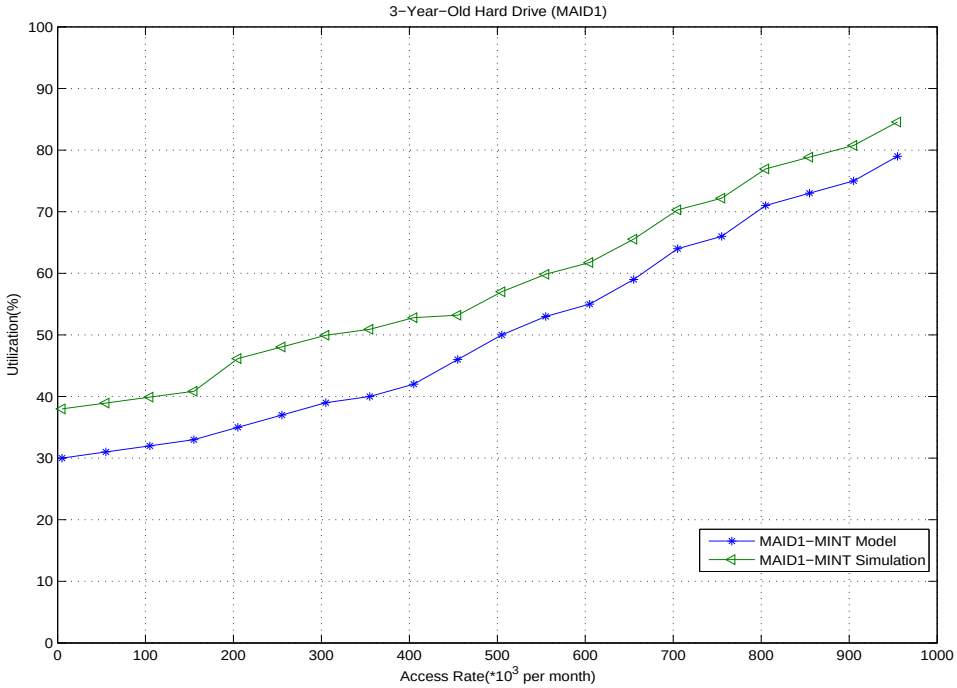

Figure 5.4: Impacts of file access rate on disk utilization (MAID1). Access rate varies from 10 to $64 * 10^4$ No./month

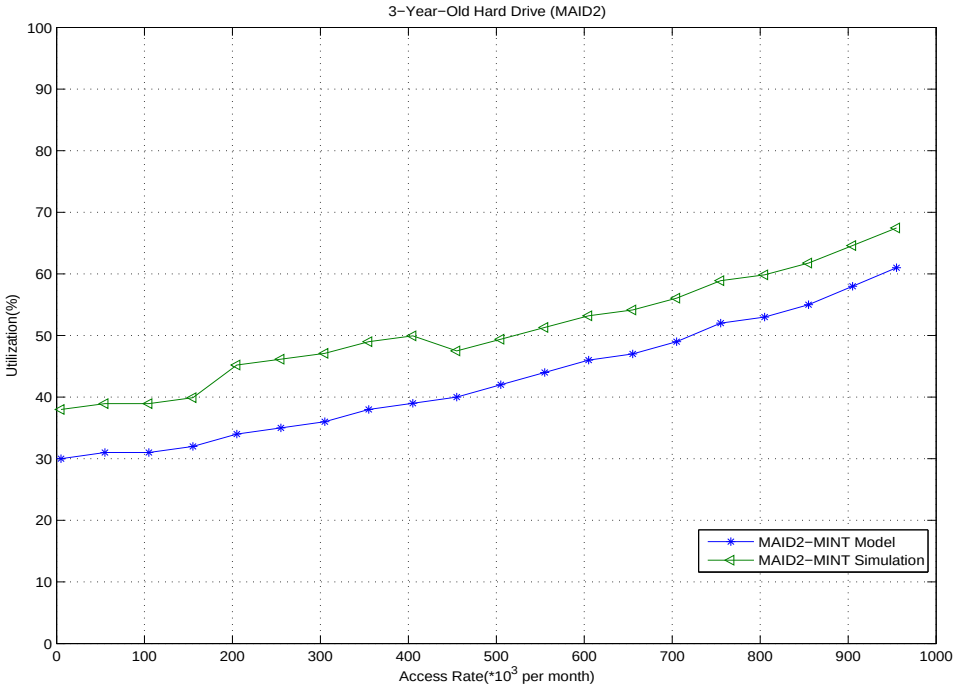

Figure 5.5: Impacts of file access rate on disk utilization (MAID2). Access rate varies from 10 to $64 * 10^4$ No./month

are noticeable differences between them when we discussed the AFR issue. Such differences are mainly due to the bath-shaped curve shown in Fig. 3.3.

5.2 Validation of MREED

5.2.1 The Validation Techniques

It is challenging to validate the accuracy of the MREED modeling framework, since we are unable to monitor PARAID running for a couple of decades. One way to address this problem is to maintain and analyze a large number of PARAID systems for a short period of time (e.g., 5 to 10 years). If one can track the systems over their entire service life, failure-rate data will be collected to validate reliability models. Even if we can test PARAID with 100 disks for five years, the sample size is small from a validation perspective.

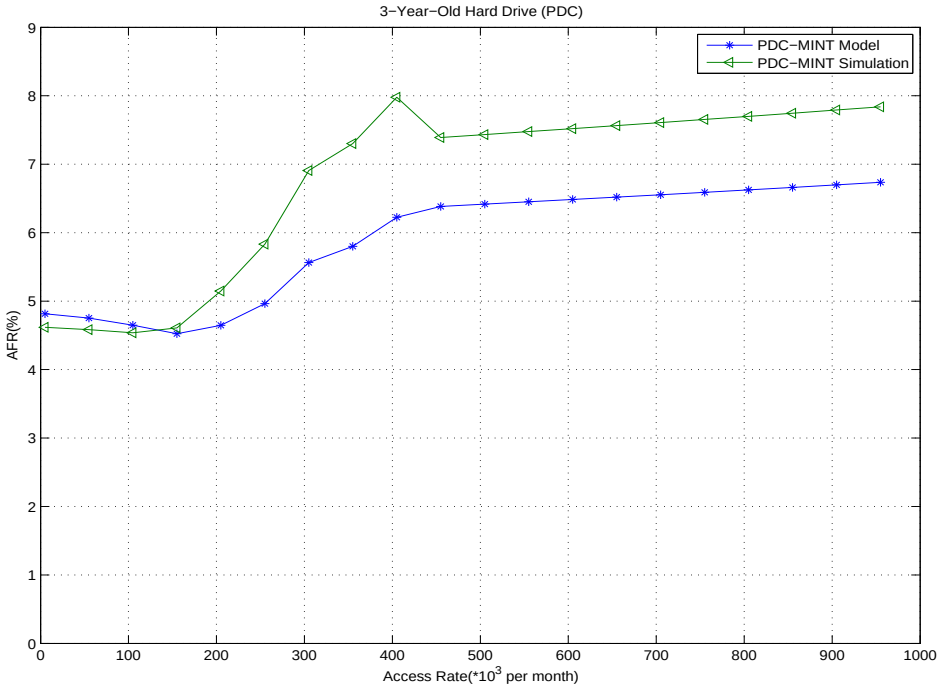

Figure 5.6: Impacts of file access rate on AFR (PDC). Access rate varies from 10 to $64 * 10^4$ No./month

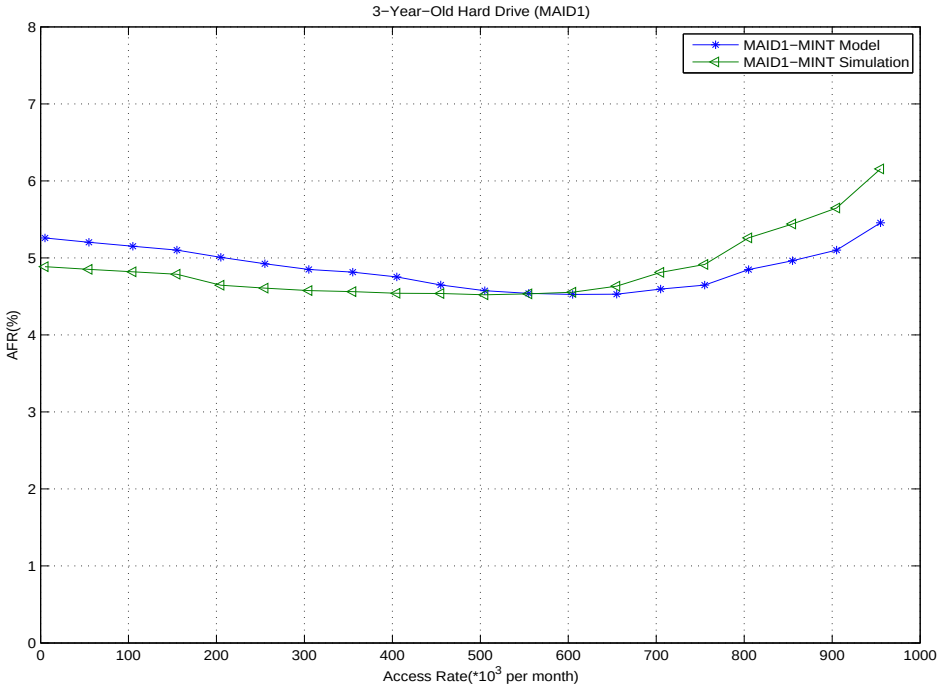

Figure 5.7: Impacts of file access rate on AFR (MAID1). Access rate varies from 10 to $64 * 10^4$ No./month

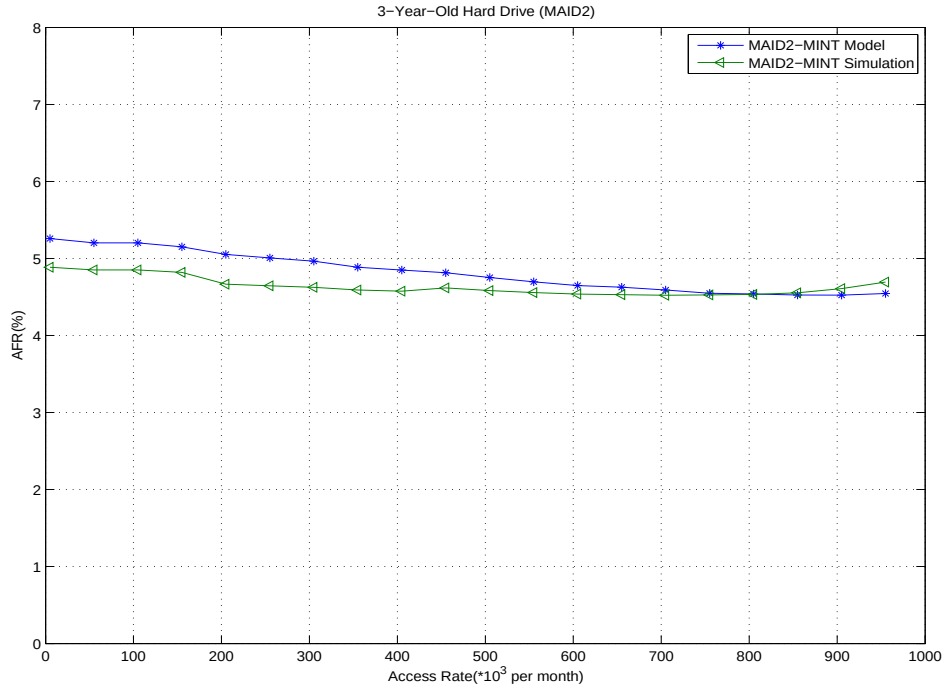

Figure 5.8: Impacts of file access rate on AFR (MAID2). Access rate varies from 10 to $64 * 10^4$ No./month

To address this validation issue, we verify MREED using the Event Validity validation technique [68], which is a practical approach to verification and validation of reliability models. Events of occurrences of our MREED model are compared to those of the widely-used storage system simulator– DiskSim– to determine if our model and DiskSim agree with one another. In our validation process, we compared a file access trace in a real-world file system

Recall that MREED consists of two major components – a utilization model and a failure-rate model. The utilization model estimates disk utilization of the PARAID system based on I/O access rates. The failure-rate model relies on Weibull distribution analysis, parameters of which were derived from a hard drive disk manufacture's report (see [20]) to predict the possibility of disk failure from its utilization.

To validate MREED, we have to validate the utilization model and the failure-rate model. Since failure rates in this study are projections based on the failure-rate

model derived from Seagate's empirical analysis (see [20]), we pay attention to the validation of the utilization model.

5.2.2 DiskSim Simulation

The DiskSim simulator, a powerful tool for the modeling and simulation of disk systems, is used widely for storage systems research [40]. Recent research projects using the DiskSim simulation environment include reducing disk I/O performance sensitivity and conserving energy in disk systems [84]. Although DiskSim is a powerful simulation tool research, there is a lack of power models in DiskSim. The Sensitivity-Based Optimization of Disk Architecture introduced accurate power models into DiskSim, but this work was based on DiskSim 2.0 [73]. Another recent study on DiskSim and power models is the Dempsey project [103]. We are grateful to the author of the EEPF papar [50] who provided us with the source code of power models developed for a newer version (i.e., version 4.0) of DiskSim. This makes it possible for us to implement utilization and power transition models into DiskSim.

5.2.3 Simulation Framework

In order to complete our validation work via DiskSim, we integrate the following two major components in the system.

- DiskSim Simulator: It is in charge of simulating the operations of all disks and data blocks managements in the sytem.
- File to Block Translator: It is responsible for mapping files residing in the storage system into block-level data.

As shown in Fig. 5.9, files are mapped into blocks before being used as inputs to the DiskSim simulator. The file-to-block converter is critical, because data blocks are typically managed within a single node and a higher level mechanism is needed

to manage data across different nodes in RAID systems. In the DiskSim simulator,

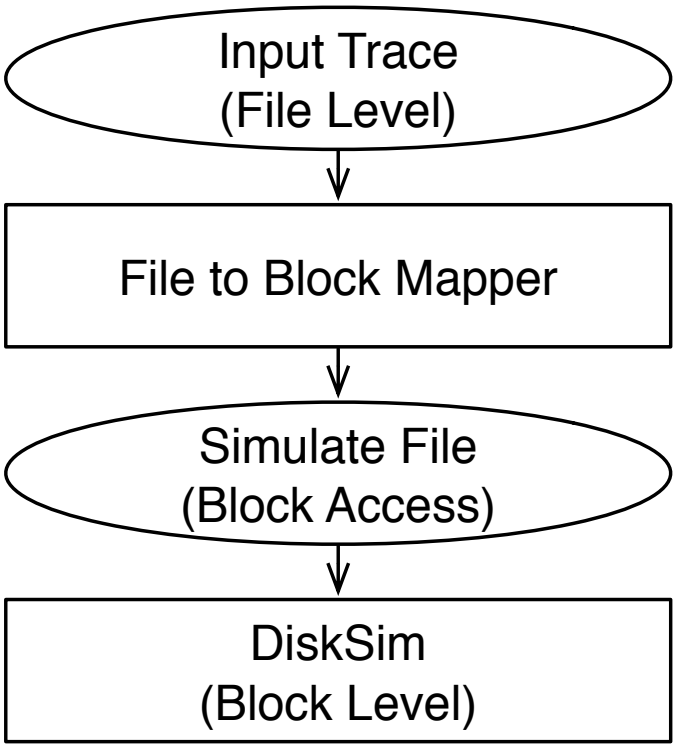

Figure 5.9: File to Block Level Converter Outline

we use the same disk model (which is a Seagate ST3146855LW hard disk drive), the I/O throughput of which is significantly high than consumer level products. In order to avoid I/O transfer throughput bottlenecks, we modify a disk architecture in the DiskSim that each single disk has its own bus and controller (see in Fig. 5.10).

5.2.4 UMass WebSearch Trace

The UMass WebSearch Trace [6] is used in the model validation process. This trace is obtained from the University of Massachusetts-Amherst (UMASS) website. The trace used in our experiments is WebSearch3.trace, which contains 4,261,709 read requests. The trace reply period is 298,715,395 milliseconds or 83 hours.

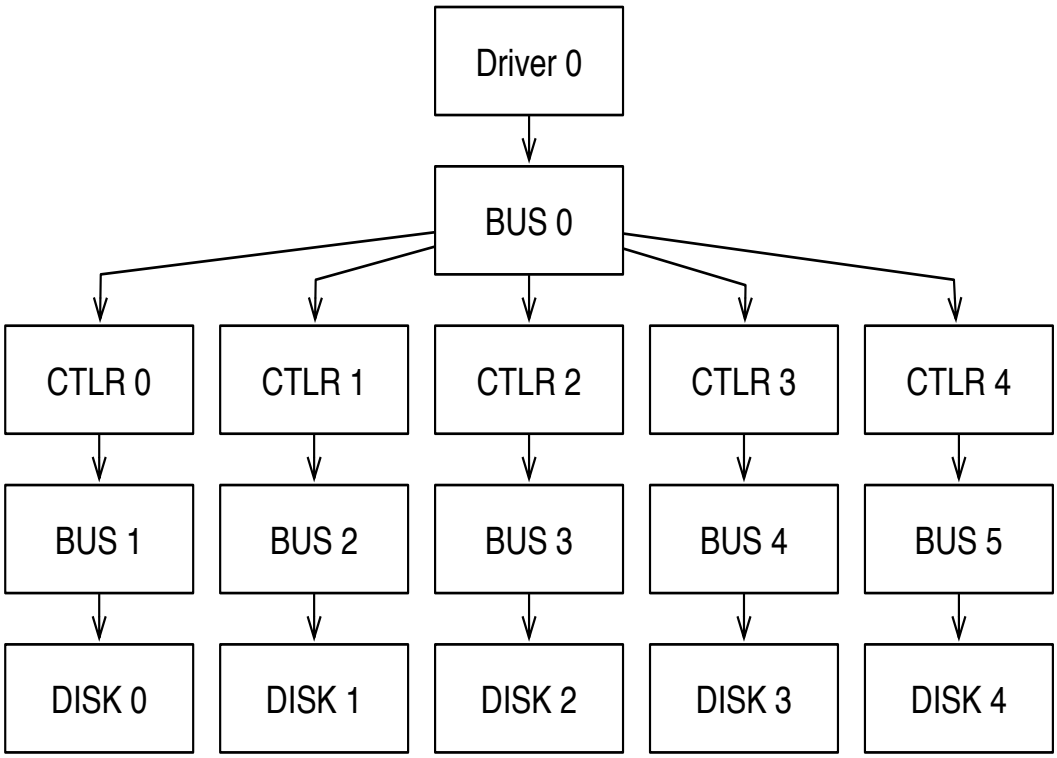

Figure 5.10: Diagram of the Storage System Corresponding to the DiskSim Raid-0

5.2.5 Validation Results

The Utilization-AFR model transfers the utilization of systems to the reliability. This model is employing the Weibull analysis by the same β and θ parameters (see Section 4.2), so we only show the validation of utilization and power transition model in this subsection.

In order to make a clearer comparison between the MREED model and the trace-driven DiskSim simulations, we divided the comparison of utilization(see Fig. 5.11) and power transition (see Fig. 5.12). We observe that results obtained from the MREED model is similar to the simulation. Furthermore, the discrepancy between the model and the simulation is below 10%.

After validating the Access Rate-Utilization sub-model, we further present the comparison results of Access Rate-Power Transition between the MREED model and the simulation results (as shown in Fig. 5.12). The figure shows that as time elapsed,

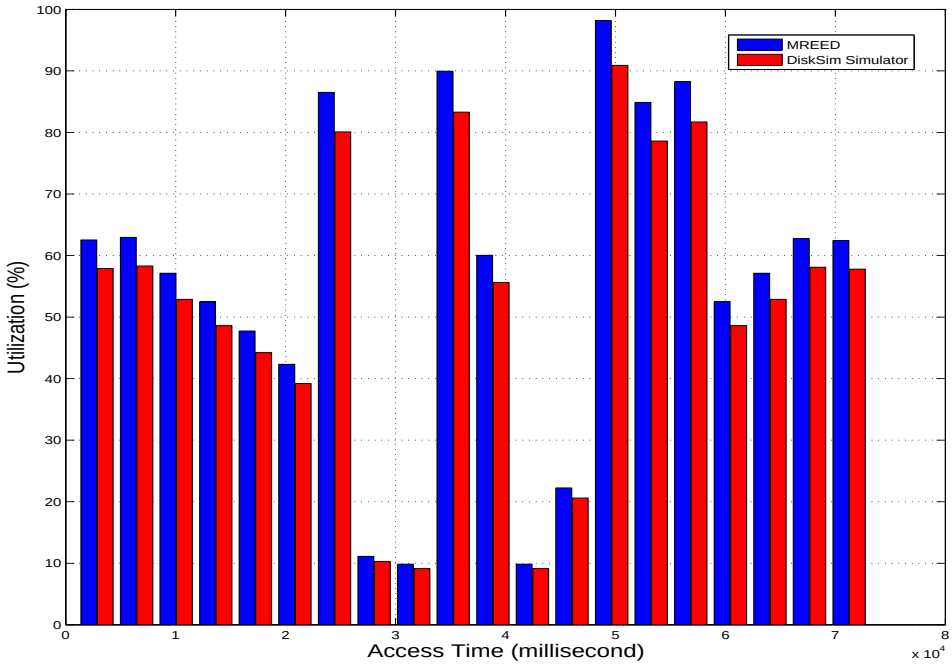

Figure 5.11: Utilization Comparison Between MREED and DiskSim Simulator

the gear shifted accordingly as files access pattern changed. Fig. 5.12 illustrates that our model performs well in estimating gear-shift events.

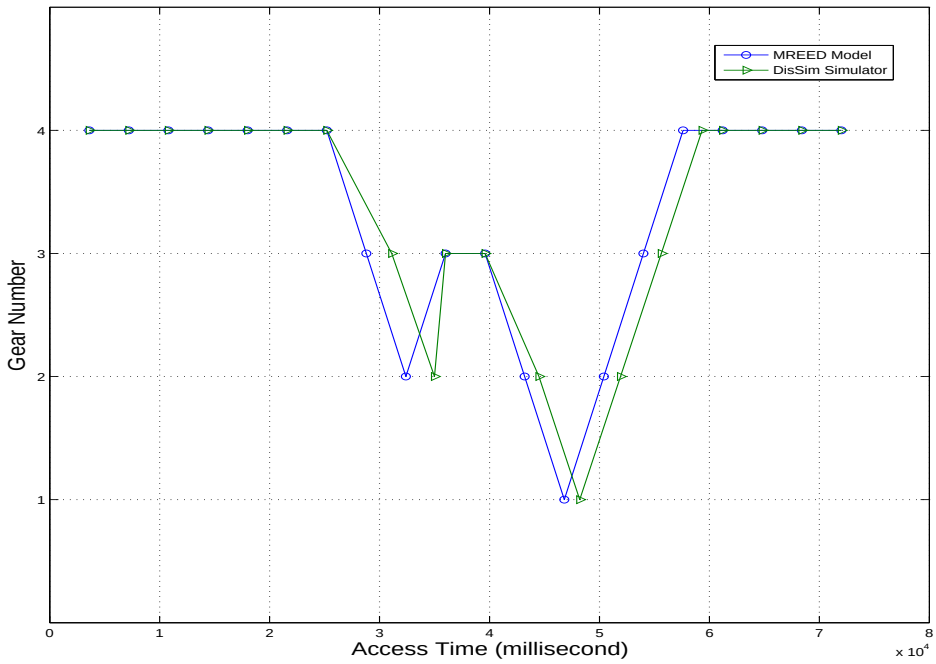

Figure 5.12: Gear Shiftings Comparison Between MREED and DiskSim Simulator

Chapter 6

Improving Reliability of Energy-Efficient Parallel Storage Systems

The Massive Array of Idle Disks (MAID) technique is an effective energy saving schemes for parallel disk systems. The goal MAID is to skew I/O load towards a few disks so that other disks can be transitioned to low power states to conserve energy. I/O load skewing techniques like MAID inherently affect reliability of parallel disks because disks storing popular data tend to have high failure rates than disks storing cold data. To achieve good tradeoffs between energy efficiency and disk reliability, we first present a reliability model to quantitatively study the reliability of energy-efficient parallel disk systems equipped with MAID schemes. Then, we propose a novel strategy—disk swapping—to improve disk reliability by alternating disks storing hot data with disks holding cold data. At Last, we further improve disk reliability by introducing multiple disk swapping strategy. We demonstrate that our disk-swapping strategies not only can increase the lifetime of cache disks in MAID-based parallel disk systems, but also further reduce the failure rate of the entire system when the multiple-disk swapping is introduced.

6.1 Introduction

Parallel disk systems, providing high-performance data-processing capacity, are of great value to large-scale parallel computers [4]. A parallel disk system comprised of an array of independent disks can be built from low-cost commodity hardware components. In the past few decades, parallel disk systems have increasingly become popular for data-intensive applications running on massively parallel computing platforms [81].

Existing energy conservation techniques can yield significant energy savings in disks. While several energy conservation schemes like cache-based energy saving approaches normally have marginal impact on disk reliability, many energy-saving schemes (e.g., dynamic power management and workload skew techniques) inevitably have noticeable adverse impacts on storage systems [12][90]. For example, dynamic power management (DPM) techniques save energy by using frequent disk spin-downs and spin-ups, which in turn can shorten disk lifetime [22] [34] [46], redundancy techniques [60] [102] [82] [89], workload skew [54] [38] [98], and multi-speed settings [32] [76]. Unlike DPM, workload-skew techniques such as MAID [19] and PDC [58] move popular data sets to a subset of disks arrays acting as workhorses, which are kept busy in a way that other disks can be turned into the standby mode to save energy. Compared with disks storing cold data, disks archiving hot data inherently have higher risk of breaking down.

Unfortunately, it is often difficult for storage researchers to improve reliability of energy-efficient disk systems. One of the main reasons lies in the challenge that every disk energy-saving research faces today, how to evaluate reliability impacts of power management strategies on disk systems. Although reliability of disk systems can be estimated by simulating the behaviors of energy-saving algorithms, there is lack of fast and accurate methodology to evaluate reliability of modern storage systems with high-energy efficiency. To address this problem, we developed a mathematical reliability model called MINT to estimate the reliability of a parallel disk system that employs a variety of reliability-affecting energy conservation techniques [99].

In this chapter, we first study the reliability of a parallel disk system equipped with a well-known energy-saving scheme— the MAID [19] technique. I/O load skewing techniques like MAID inherently affect reliability of parallel disks because of two reasons: First, disks storing popular data tend to have high I/O utilization than disks storing cold data. Second, disks with higher utilization are likely to have higher risk

of breaking down. To address the adverse impact of load skewing techniques on disk reliability, a disk swapping strategy was proposed to improve disk reliability in MAID by switching the roles of data disks and cache disks. We evaluate impacts of the disk swapping scheme on the reliability of MAID-based parallel disk systems.

We summarize our contributions as follows:

1. We developed a model for Massive Array of Idle Disks (MAID) based on Mathematical Reliability Models for Energy-efficient Parallel Disk System (MINT) [99];

2. We built single disk swapping and multiple disk swapping mechanisms to improve reliability of various load skewing techniques.

3. We studied the impacts of the disk swapping schemes on the reliability of MAID.

The remainder of this chapter is organized as follows. Section 6.2 studies single disk swapping and multiple disks swapping strategies on MAID. Section 6.3 presents experimental results and performance evaluation. Finally, Section 6.4 concludes the chapter with discussions.

6.2 Improving Reliability of MAID via Disk Swapping

6.2.1 Improving Reliability of Cache Disks in MAID

Cache disks in MAID are more likely to fail than data disks due to the two reasons. First, cache disks are always kept active to maintain short I/O response times. Second, the utilization of cache disks is expected to be much higher than that of data disks. From the aspect of data loss, the reliability of MAID relies on the failure rate of data disks rather than that of cache disks. However, cache disks tend to be a single point of failure in MAID, which if the cache disks fail, will stop MAID from conserving energy. In addition, frequently replacing failed cache disks can increase hardware and management costs in MAID. To address this single point

of failure issue and make MAID cost-effective, we designed a disk swapping strategy for enhancing the reliability of cache disks in MAID.

Fig. 6.1 shows the basic idea of the disk swapping mechanism, according to which disks rotate to perform the cache-disk functionality. In other words, the roles of cache disks and data disks will be periodically switched in a way that all the disks in MAID have equal chance to perform the role of caching popular data. For example, the two cache disks on the left-hand side in Fig. 6.1 are swapped with the two data disks on the right-hand side after a certain period of time (see Section 6.3.3 for circumstances under which disks should be swapped). For simplicity without losing generality, we assume that all the data disks in MAID initially are identical in terms of reliability. This assumption is reasonable because when a MAID system is built, all the new disks with the same model come from the same vendor. Initially, the two cache disks in Fig. 6.1 can be swapped with any data disk. After the initial phase of disk swapping, the cache disks are switched their role of storing replica data with the data disks with the lowest annual failure rate. In doing so, we ensure that cache disks are the most reliable ones among all the disks in MAID after each disk swapping process. It is worth noting that the goal of disk swapping is not to increase mean time to data loss, but is to boost mean time to cache-disk failure by balancing failure rates across all disks in MAID.

Fig. 6.2 is the logic diagram of the single disk swapping mechanism, which demonstrates more details about the swapping. When the access rate reaches the threshold, which is set beforehand, a data disk's capacity will be checked. If the data disk has enough free space to hold all the replicas that are hold by a cache disk, it will be paired with the cache disk for swapping later. Otherwise, other data disks' capacity will be checked until a disk that meets the requirement. If there is no disk meets the requirement, the disk swapping won't be executed. This step needs to be executed first to prevent the original data from miss-deleting on the data disk. In our research,

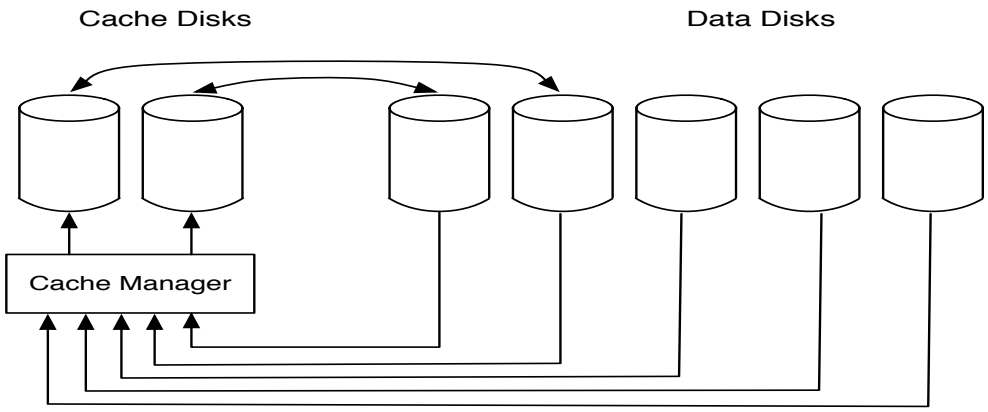

Figure 6.1: Disk Swapping in MAID: The two cache disks on the left-hand side are swapped with the two data disks on the right-hand side

we assumed that the data disk's capacity is large enough to hold all the cache data and to keep the original data. The capacity of the cache disk will be examined when it is paired with a data disk.

If the cache disk has enough free space to hold all the data that are hold by the data disk, the data disk will duplicate all the cache data from the cache disk while holding all the original data. Then the cache disk will copy the data from the data disk and keeps all replicas of its own. On the other hand, if the cache disk does not have enough free space to hold all the data from the data disk, all replicas it holds will be deleted after they are duplicated to the destination releasing the space for the data copied from the data disk. At this step, no matter the cache disk has available capacity or not, the data needs to be transfered from cache disk first to prevent original data from either miss-deleting or losing.

Algorithm 1 outlined below is the single-disk-swapping algorithm that switches the roles of cache disks and data disks to improve the reliability of cache disks. The algorithm is called single-disk-swapping because the disk swapping occurs only once in MAID.

Disk swapping is very beneficial to MAID for two reasons. First, disk swapping further improves the energy efficiency of MAID because any failed cache disk

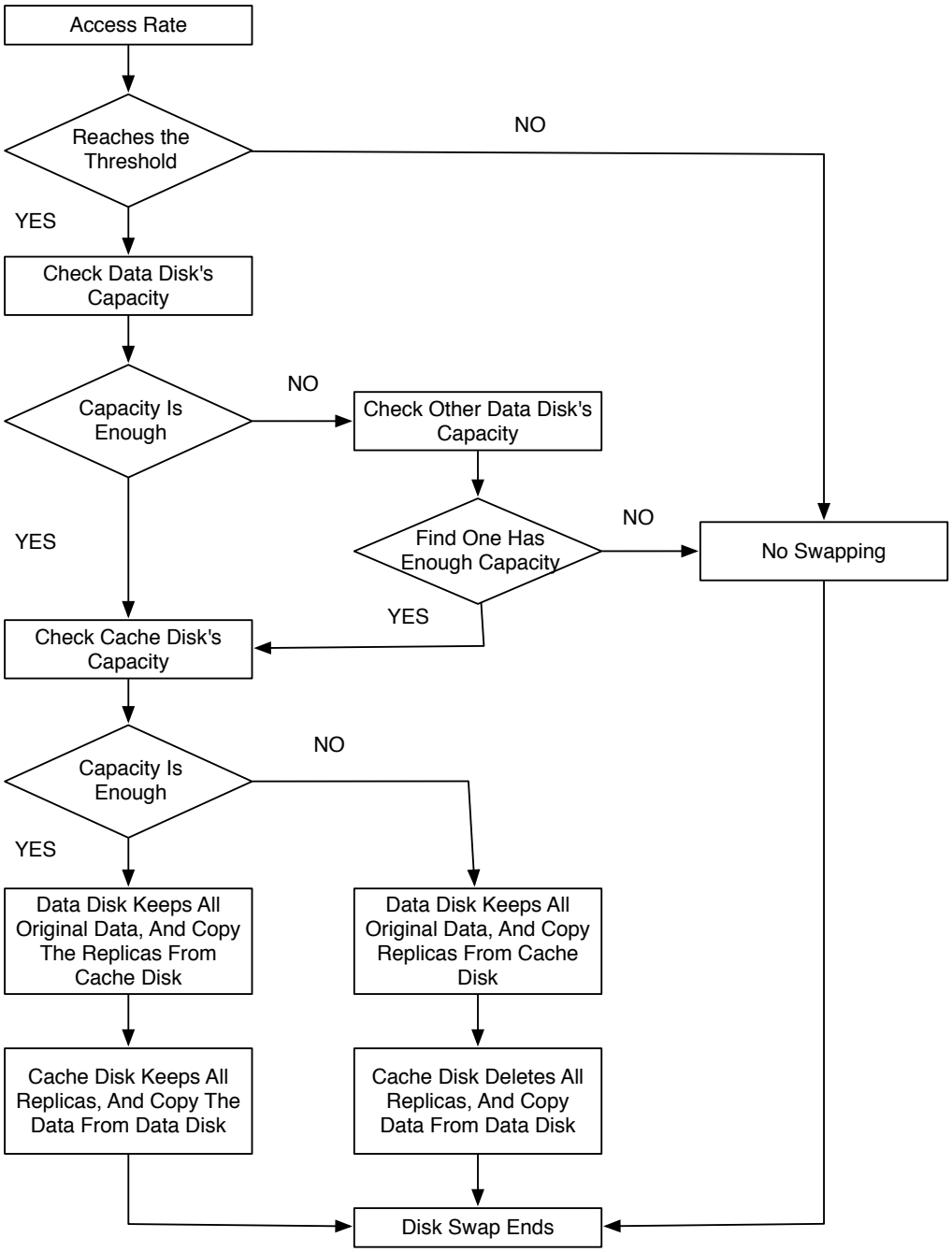

Figure 6.2: Logic Diagram of Disk Swapping

Algorithm 1 The Single-Disk-Swapping Algorithm

```
1: Input The Access Rate of The System;
2: if The Access Rate Reaches The Threshold then
3:    Check the Available Capacity of Data Disk;
4:    if The Available Capacity of Data Disk Is Enough then
5:       Check the Available Capacity of Cache Disk;
6:       if The Available Capacity of Cache Disk Is Enough then
7:          Data Disk Keeps All Original Data and Duplicates Cache Data From Cache
            Disk;
            Cache Disk Keeps All Replicas and Copies Data From Data Disk;
8:       else
9:          if The Available Capacity of Cache Disk Is NOT Enough then
10:            Data Disk Keeps All Original Data and Duplicates Cache Data From
               Cache Disk;
               Cache Disk Deletes All Replicas and Copies Data From Data Disk;
11:         end if
12:      end if
13:   else
14:      if The Available Capacity of Data Disk Is NOT Enough then
15:         while There Is A Data Disk That Has Enough Available Capacity do
16:            Check the Available Capacity of Cache Disk;
17:         end while
18:      end if
19:   end if
20: else
21:   Don't Do Swap;
22: end if
23: Disk Swap Ends;
```

can prevent MAID from effectively saving energy. Second, disk swapping reduces maintenance cost of MAID by making cache disks less likely to fail.

6.2.2 Swapping Disks Multiple Times

Now we consider the case where disk swapping is invoked multiple times in MAID. As described in Section 6.2.1, the single-disk-swapping mechanism improves the reliability of the MAID system by making all disks have equal chance to perform the role of cache disks that have high I/O workload and high utilization. The single-disk-swapping algorithm has a major limitation, because disks are swapped only once throughout their lifetimes. That means single-disk-swapping only affects the reliability for a very short period of time. After each disk swapping, the utilization of those disks with low AFRs are likely to be kept at a high level, which in turn leads to an increasing AFR of the entire disk system. In order to improve the reliability of the MAID system for a long time period (e.g., 1,000,000 hours or over 100 years [71]), we address the issue of swapping disks multiple times (see multiple disk swapping shown in Algorithm 2).

In the multiple-disk-swapping algorithm, the number of disk-swapping per month is an important parameter affecting both reliability and performance of MAID. This parameter can either be manually set as a constraint or be configured dynamically according to changing workload conditions. In the static approach, the disk-swapping mechanism is triggered after MAID has been operating for a certain number of days regardless I/O workload. For example, if the frequency is set as three times per month, disks will be swapped once every ten days.

In the dynamic approach, the disk-swapping function is invoked once workload conditions (i.e., access rate) meet the configured value regardless the time intervals between two swaps. For instance, if the access rate is set as $2*10^5$ Numbers per month, the disks will be swapped every time when the access rate reaches $2 * 10^5$No./Month.

The dynamic multiple-disk-swapping scheme ensures that disk swaps occur only when it is necessary.

Algorithm 2 The Algorithm for Multiple Disk Swapping

```
1: while The Frequency of Disk Swapping Is No More Than The Given Ones do
2:    Run Algorithm 1
3: end while
4: Disk Swap Ends;
```

6.3 Experimental Results and Evaluation

6.3.1 Experimental Setup

We developed a simulator to validate the reliability model for MAID. It might be unfair to compare the reliability of MAID with any non-energy-efficient parallel disks, since MAID trades extra cache disks for high energy efficiency. To make fair comparisons, we considered a MAID system with two configurations. The first configuration referred to as MAID-1 employs existing disks in a parallel disk system as cache disks to store frequently accessed data. Thus, the first configuration of MAID improves energy efficiency of the parallel disk system at the cost of capacity. In contrast, the second configuration— called MAID-2—needs extra disks to be added to the disk system to serve as cache disks.

Our experiments were started by evaluating the reliability of the original MAID system without disk swapping. Then, we studied the reliability impacts of the single-disk-swapping strategy on MAID. Finally, we assessed the reliability impacts of the multiple-disk-swapping scheme. We simulated MAID-1, and MAID-2 coupled with the disk-swapping strategies in two parallel disk systems described in Table 6.1. For the MAID-1 configuration, there are 5 cache disks and 15 data disks. In the disk system for the MAID-2 configuration, there are 5 cache disks and 20 data disks. As for the case of PDC, we fixed the number of disks to 20. Thus, we studied MAID-2

Table 6.1: The characteristics of the simulated parallel disk system used to evaluate the reliability of MAID-1, and MAID-2.

Energy-efficiency Scheme	Number of Disks	File Access Rate (No. per month)	File Size (KB)
NONE*	20 data (20 in total)	0˜10^6	300
MAID-1	15 data+5 cache (20 in total)	0˜10^6	300
MAID-2	20 data+5 cache (25 in total)	0˜10^6	300

Original Disk System Without Any Energy-Efficiency Scheme

and PDC using a parallel disk system with 20 disks; we used a similar disk system with totally 25 disks to investigate MAID-1. We varied the file access rate in the range between 0 to 10^6 times per month. The average file size considered in our experiments is 300KB. The base operating temperature is set to 35°C. In this study, we focused on read-only workload. Nevertheless, the MINT model should be readily extended to capture the characteristics of read/write workloads.

6.3.2 Disk Utilization

Fig. 6.3 shows that when the average file access rate increases, the utilizations of MAID-1 and MAID-2 increase accordingly. Compared with the utilization of MAID-2, the utilization of MAID-1 is more sensitive to the file access rate. Under low I/O load, the utilizations of MAID-1 and MAID-2 are very close to each other. When I/O load becomes relatively high, the utilization of MAID-1 is slightly higher than that of MAID-2. This is mainly because the capacity of MAID-2 is larger than that of MAID-1.

6.3.3 The Single-Disk-Swapping Strategy

A key issue of the disk-swapping strategies is to determine circumstances under which disks should be swapped in order to improve disk system reliability. One

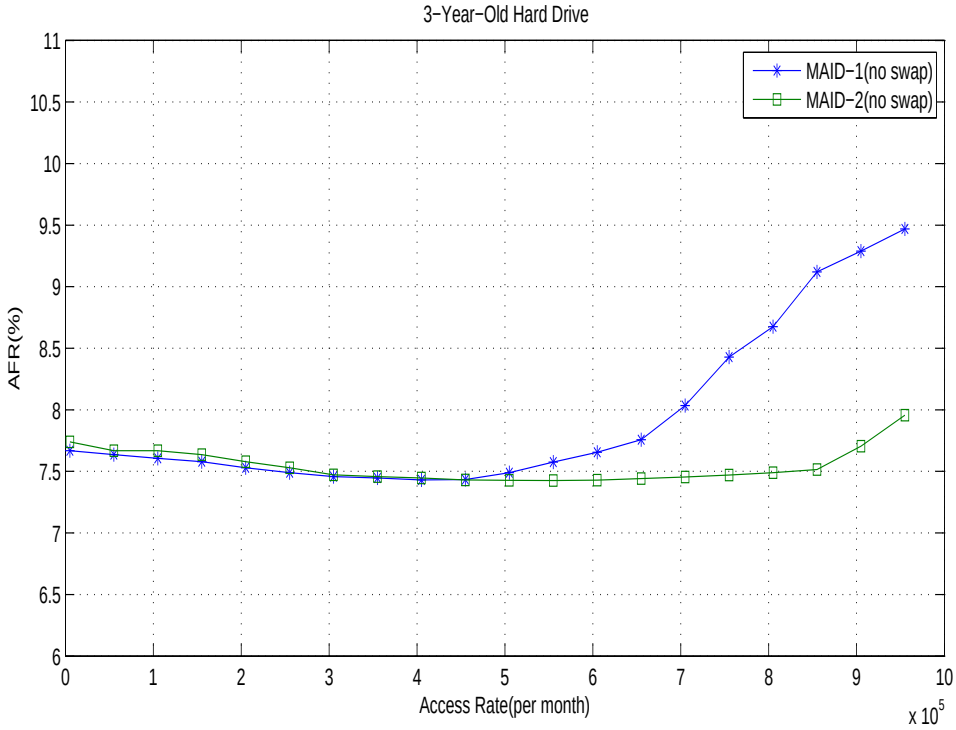

Figure 6.3: Utilization Comparison of the MAID
Access Rate Impacts on AFR (No Swapping)

straightforward way to address this issue is to periodically initiate the disk-swapping process. For example, we can swap disks in MAID once every month. Periodically swapping disks, however, might not always enhance the reliability of parallel disk systems. For instance, swapping disks under very light workloads cannot substantially improve disk system reliability. In some extreme cases, swapping disks under light workload may worsen disk reliability due to overhead of swapping. As such, our disk-swapping strategies do not periodically swap disks. Rather, the disk-swapping process is initiated when the average I/O access rates exceed a threshold. In our experiments, we evaluated the impact of this access-rate threshold on the reliability of a parallel disk system. More specifically, the threshold is set to $2 * 10^5$, $5 * 10^5$, and $8 * 10^5$ times/month, respectively. These three values are representative values for the threshold because when the access rate hits $5 * 10^5$, the disk utilization lies

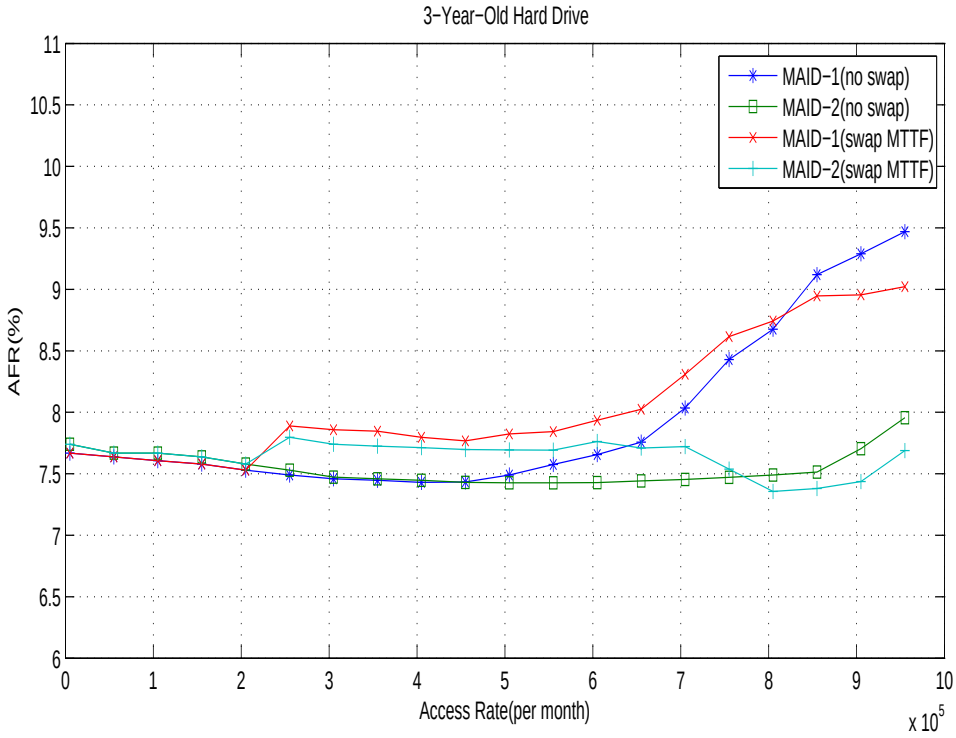

Figure 6.4: Utilization Comparison of the MAID
Access Rate Impacts on AFR (Threshold=$2 * 10^5$)

in the range between 80% and 90% [61], which in turn ensures that AFR increases with the increasing value of utilization (see Fig. 3.7).

Figs. 6.4, 6.5, and 6.6 reveal the annual failure rates (AFR) of MAID-1 and MAID-2 with and without using the proposed disk-swapping strategy. The results plotted in Figs. 6.4, 6.5, and 6.6 show that for both MAID-1 and MAID-2, the disk-swapping process reduces the reliability of data disks in the disk system. We attribute the reliability degradation to the following reasons. MAID-1 and MAID-2 only store replicas of popular data; the reliability of the entire disk system is not affected by failures of cache disks. The disk-swapping processes increase the average utilization of data disks, thereby increasing the AFR values of data disks. Nevertheless, the disk-swapping strategy has its own unique advantage. Disk swapping is intended to reduce hardware maintenance cost by increasing the lifetime of cache disks. In other

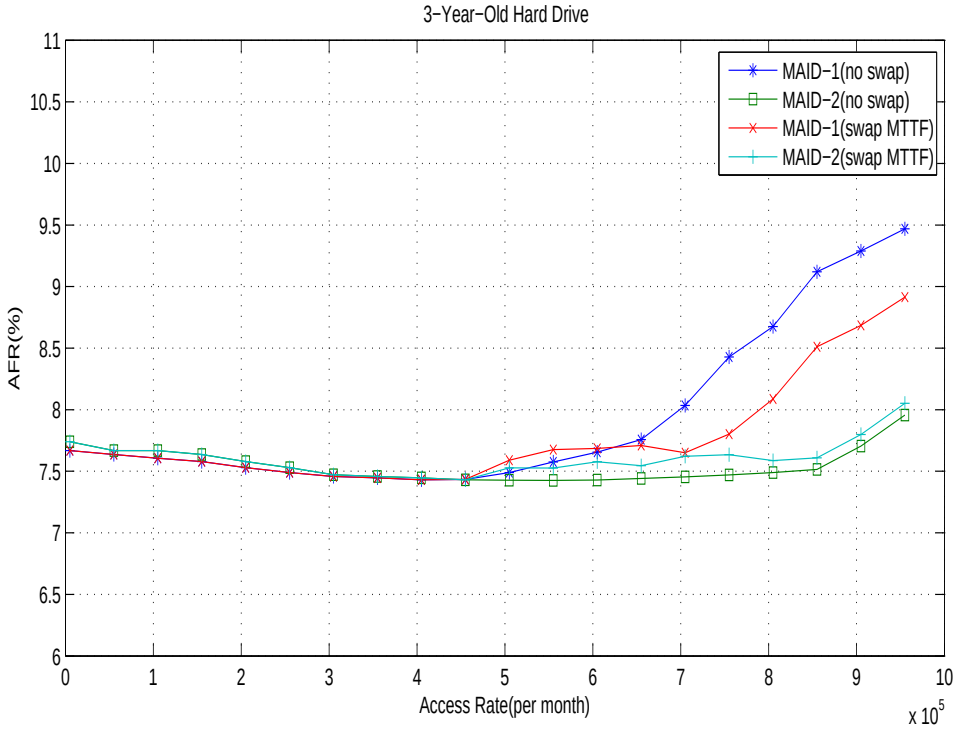

Figure 6.5: Utilization Comparison of the MAID
Access Rate Impacts on AFR (Threshold=$5 * 10^5$)

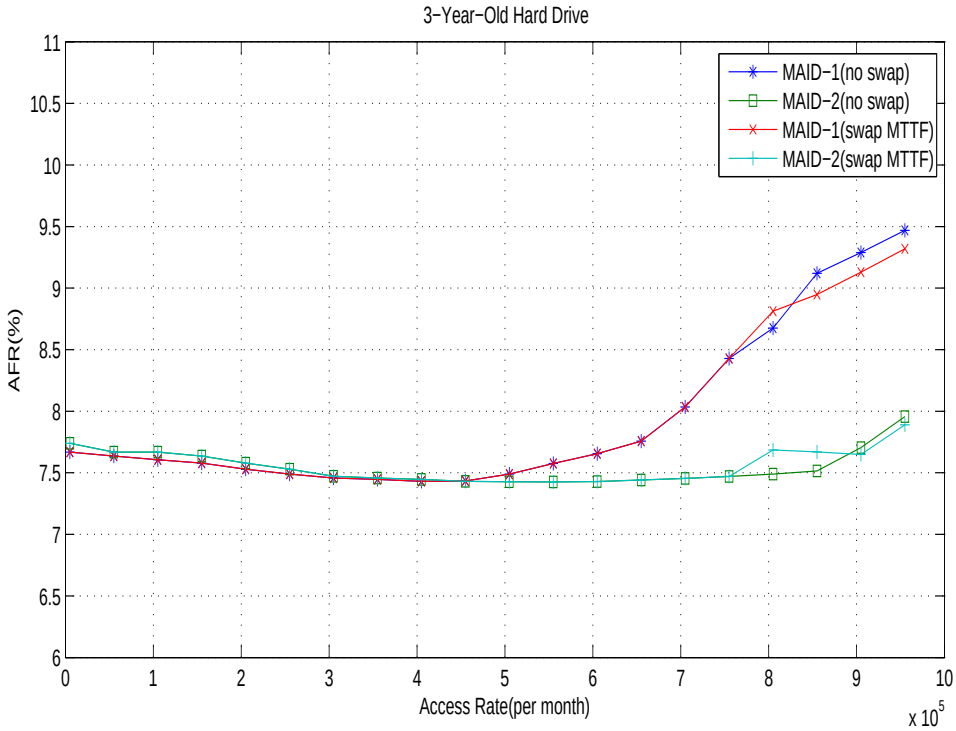

Figure 6.6: Utilization Comparison of the MAID
Access Rate Impacts on AFR (Threshold=$8 * 10^5$)

words, disk swapping is capable of extending the Mean Time To Failure or MTTF [61] of the cache disks.

We observed from Figs. 6.4, 6.5, and 6.6 that for the MAID-based disk system with the disk-swapping strategy, a small threshold leads to a low AFR. Compared with the other two thresholds, the $2 * 10^5$ threshold showed in Fig. 6.4 results in the lower AFR. The reason is that when the access rate is $2 * 10^5$ No./month, the disk utilization is around 35% [61], which lies in the monotone decreasing area of the curve shown in Fig. 3.7. Thus, disk swapping reduces AFR for a while until the disk utilization reaches 60%.

6.3.4 The Multiple-Disk-Swapping Strategy

Section 6.3.3 shows that single-disk-swapping strategy can improve the reliability of the MAID system. However, the single-disk-swapping has minimal reliability impact in a long period of time. For example, Fig. 6.4 indicates that after swapping cache and data disks, the failure rate of the disk system continues going up as the access rate keeps increasing. We observed that after the first disk swap without any consecutive disk swaps, the failure rate of disk-swapping-enabled MAID will become close to that of non-disk-swapping MAID. Thus, disk swapping must be repeatedly conducted under the condition that the failure rate of MAID increases.

To evaluate the multiple-disk-swapping scheme, we configured the access rate threshold to $2 * 10^5$, $2.5 * 10^5$, and $4 * 10^5$ No./month. For example, if the threshold is set to $2 * 10^5$, the total access rate can be as high as $8 * 10^5$, which is one of the thresholds chosen for the single-disk-swapping strategy.

Figs. 6.7, 6.8, and 6.9 reveal the annual failure rates (AFR) of MAID-1 and MAID-2 with both a single disk swap and multiple disk swaps. The results show that the multiple-disk-swapping process further reduces the failure rate of data disks in the MAID system. Comparing the AFR values plotted in Figs. 6.4, 6.5, and 6.6,

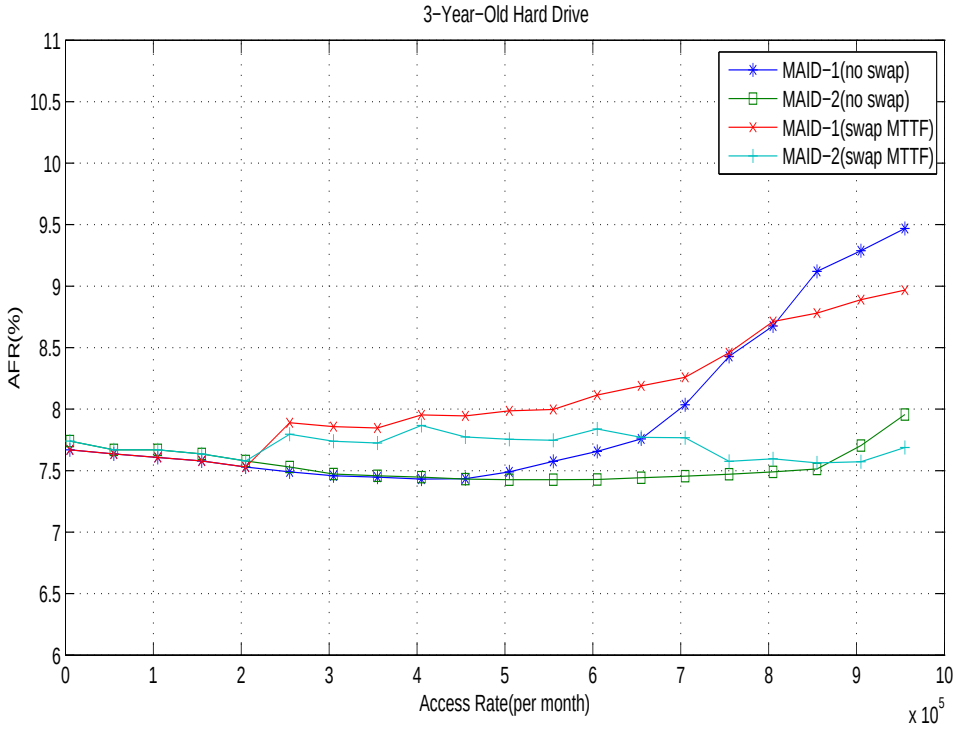

Figure 6.7: Utilization Comparison of the MAID
Access Rate Impacts on AFR (Multiple Threshold=$2 * 10^5$)

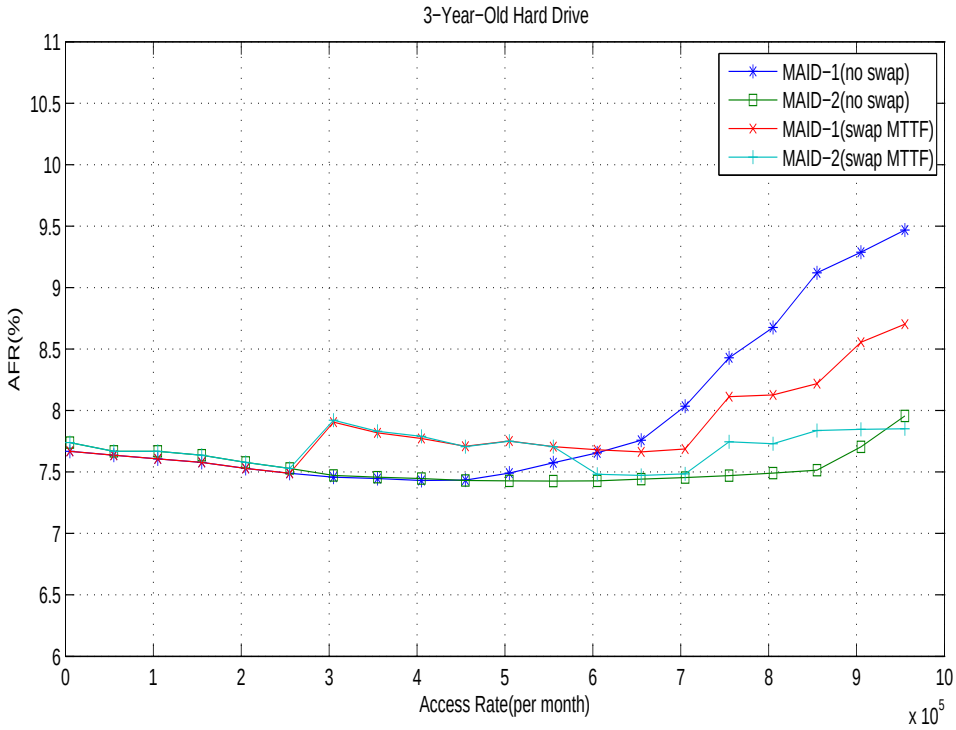

Figure 6.8: Utilization Comparison of the MAID
Access Rate Impacts on AFR (Multiple Threshold=$2.5 * 10^5$)

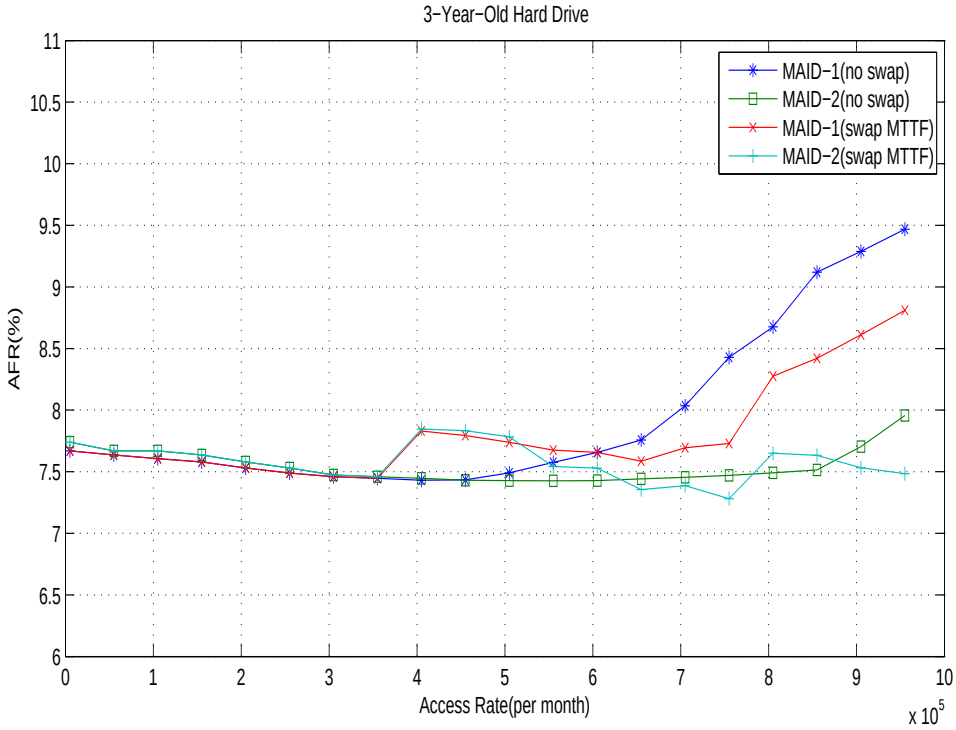

Figure 6.9: Utilization Comparison of the MAID
Access Rate Impacts on AFR (Multiple Threshold=$4 * 10^5$)

we noticed that the failure rate of MAID with multiple disk swaps is lower than that of the same with with a single disk swap at access rate $10 * 10^5$. As the access rate increases, the reliability improvement achieved by the multiple-disk-swapping scheme becomes more pronounced. The major reason behind the improvement is that swapping disks multiple times can continue balancing I/O workload of each disk in the MAID system in the long run. After each disk swap, if the failure rate of MAID increases to a certain point, (see, for example, Fig. 6.3) a subsequent disk swap will be initiated.

Figs. 6.7, 6.8, and 6.9 demonstrate that the failure rate of the multi-swapping MADI system changes periodically. For exampple, Fig. 6.7 shows that immediately after each disk swapping process, the failure rate of MAID increases 5% due to the overhead caused by copying data among cache disks and data disks. Then, the failure rate stays stable for a while until the next disk swapping occurs. We observe that at

the second disk swap, the cumulative access rate is $4 * 10^5$, which is the same as the first swapping threshold shown in Fig. 6.9. The forth disk-swapping point in Fig. 6.7 is the same as that single disk swapping threshold shown in Fig. 6.6. Comparing Fig. 6.9 and Fig. 6.6, we conclude that when access rate reaches $10 * 10^5$, the failure rate of the multiple-disk-swapping scheme is lower than that of the single-disk-swapping scheme. This reliability improvement is made possible by multiple disk swaps, because cache disks and data disks are switched after the failure rates of the cache disks become higher than those of the data disks. Repeatedly swapping cache and data disks can well balance the failure rates of all the disks in the MAID system.

6.4 Summary

This chapter presents a reliability model to quantitatively study the reliability of energy-efficient parallel disk systems equipped with the Massive Array of Idle Disks (MAID) technique. Note that MAID is a well-known effective energy-saving schemes for parallel disk systems. It aims to skew I/O load towards a few disks so that other disks can be transitioned to low power states to conserve energy. I/O load skewing techniques like MAID inherently affect reliability of parallel disks because disks storing popular data tend to have high failure rates than disks storing cold data. To address the reliability issue in MAID, we developed single disk-swapping strategies to improve disk reliability by alternating disks storing hot data with disks holding cold data. Additionally, we introduced multiple disk-swapping scheme to further improve reliability of MAID. Then we quantitatively evaluated the impacts of the disk-swapping strategies on reliability of MAID-based disk systems. We demonstrated that the disk-swapping strategies not only can increase the lifetime of cache disks in MAID-based parallel disk systems, but also can improve its reliability in the long period of time by balancing the workload of cache disks and data disks then balancing the their utilization correspondingly.

Future directions of this research can be performed in the following. First, we will extend the MINT model to investigate mixed read/write workloads in the future. Second, we will investigate a fundamental trade-off between reliability and energy-efficiency in the context of energy-efficient disk arrays. A tradeoff curve will be used as a unified framework to justify whether or not it is worth trading reliability for high energy efficiency. Last, we will study the most appropriate conditions under which disk-swapping processes should be initiated.

Chapter 7

Conclusion and Future Work

7.1 Main Contributions

7.1.1 The MINT model for parallel storage systems

In recognition that existing disk reliability models cannot be used to evaluate reliability of energy-efficient disk systems, we propose a new model called MINT to evaluate the reliability of a disk array equipped with reliability-affecting energy conservation techniques. We first model the impacts of disk utilization and power-state transition frequency on reliability of each disk in a disk array. We then derive the reliability of an individual disk from its utilization, age, temperature, and power-state transition frequency. Finally, we use MINT to study the reliability of disk arrays coupled with the MAID (Massive Array of Idle Disks) technique and the PDC (Popular Disk Concentration technique) technique.

7.1.2 The MREED model for RAID systems

We presente a reliability model called MREED to quantitatively study the reliability of energy-efficient parallel disk systems equipped with the PARAID technique. Note that PARAID is a newly developed energy-saving scheme for RAID systems. It aims to skew I/O load towards a few disks so that other disks can be transitioned to low power states to conserve energy. I/O load skewing techniques like PARAID inherently affect reliability of RAID disks, because disks keep working on low gears tend to have high failure rates, let alone the risk of failure caused by data duplicating during the gear shifting. Furthermore, once the number of failed disks exceeds the

systems tolerance, data in the system are lost without any chance of being recovered. To address the model validation issue for MREED, we modified the DiskSim simulator, which is a widely-used storage system simulator, to validate our access-rate-utilizaiton sub-model of MREED by comparing the utilization of 5-disk PARAID system using a real-world disk I/O trace with the utilization that calculated from the MREED model using the same trace.

7.1.3 Reliability improvement of parallel storage systems

This dissertation presents a reliability model to quantitatively study the reliability of energy-efficient parallel disk systems equipped with the Massive Array of Idle Disks (MAID) technique. Note that MAID is a well-known effective energy-saving schemes for parallel disk systems. It aims to skew I/O load towards a few disks so that other disks can be transitioned to low power states to conserve energy. I/O load skewing techniques like MAID inherently affect reliability of parallel disks because disks storing popular data tend to have high failure rates than disks storing cold data. To address the reliability issue in MAID, we develop single disk-swapping strategies to improve disk reliability by alternating disks storing hot data with disks holding cold data. Additionally, we introduce multiple disk-swapping scheme to further improve reliability of MAID. Then we quantitatively evaluate the impacts of the disk-swapping strategies on reliability of MAID-based disk systems. We demonstrate that the disk-swapping strategies not only can increase the lifetime of cache disks in MAID-based parallel disk systems, but also can improve its reliability in the long period of time by balancing the workload of cache disks and data disks then balancing the their utilization correspondingly.

7.2 Future Work

7.2.1 Future Directions for the Short Term

Our short-term interest will concentrate on the following two directions, which are the extensions of my past and current research on reliability analytical model for parallel storage systems

- **Fault Tolerance Analysis for RAID Storage Systems**

  Although the MINT model presented in this dissertation is adequate to quantify the reliability of energy-efficient disk arrays, MINT is insufficient to analyze energy-ware RAID systems. We plan to investigate a more sophisticated model that can modify data access patterns and the stripped data placement. To reduce power, a conventional RAID system cannot simply rely on caching and powering off disks during idle periods due to its disk parallelism–all disks are spinning even under a light load. By varying the number of powered-on disks via *gear-shifting* or switching among sets of disks (e.g. Power-Aware Redundant Array of Inexpensive Disks), the energy consumption of a RAID system can be reduced. However, after changing the number of active disks in the system, the RAID level will be changed accordingly. This affects the reliability of the system. As a further extension of this dissertation, we plan to investigate the behavior of RAID levels in terms of gear shifting and the striped data movement along with input data access patterns.

- **Predictive Reliability Models for Storage Systems**

  Reliability evaluation of a disk system indicates the present liability of the system. We argue that if one can predict the reliability of a storage system, the system's maintenance expenses can be reduced as disks will be replaced on time. Risks that disks will fail before being replaced can be diminished and the chances of purchasing new disks can be decreased. The goal of this future

research is to build up a predictive reliability models to forecast reliability of storage systems based on data access patterns and to provide disks maintenance suggestions. Furthermore, such a strategy can be integrated with load balancing schemes to ensure tow policies. First, disks reaching the end of their lifetimes will be assigned with lighter workloads. Second, data on disks that are likely to fail will be backed-up in right time.

7.2.2 Future Directions for the Long Term

We plan to pursue the following three long-term research goals.

- **Energy-Aware Storage Systems in Data Centers**

  Distributed File Systems are becoming the de-facto method of data storage for the new generation of data centers (e.g., web applications by companies like Google, Amazon, and Yahoo!). There are several reasons that distributed storage mechanisms are preferred over traditional relational database systems including scalability, availability and performance. However, the energy consumption issue needs to be addressed carefully in data centers. For example, a 360-T flops supercomputer (e.g., IBM Blue Gene/L) with traditional processors needs 2,329.60KW/h to be operated. This energy requirement is approximately equal to the sum of 22,000 US households' energy consumption. In addition, high-temperature heat dissipation caused by large-scale clusters requires cooling equipments (e.g., air conditioners) to control temperatures in supercomputer and data centers.The trends in power/cooling delivery and cost highlight the need for support in data centers for power and thermal management. In the long term, we plan to explore schemes in utilizing platform power management(e.g., processor frequency scaling, prefetching, caching, data management, load balancing, etc) for data centers.

- **Reliability-Aware Parallel Virtual File System(PVFS) in High-Performance Computing**

  PVFS, a popular network clustering file system, brings state-of-the-art parallel I/O concepts to production parallel systems. It is designed to scale to petabytes of storage and provide access rates at 100s of GB/s. While working on a PVFS-related research project, we realized that the energy-saving may not be a central issue for high-performance computing(HPC) systems. One of the major reasons is that energy-efficiency schemes usually negatively affect to the main goal of a HPC system, which aims to maximized system performance. However, the fault-tolerant issue plays an important role in HPC systems, because any minor defect may cause data tragedies of the entire system. Hence, we plan to develop fault tolerant mechanism for PVFS in order to enhance availability.

- **Information Assurance and Security in Cloud Storage Systems**

  Providing confidentiality, integrity, authenticity, privacy and availability of information are essential for the normal operation in cloud computing. Hence, information assurance and security is a critical issue. As the last long-term research direction, we will place emphasis on schemes of authorization and authentication for cloud storage systems

Bibliography

[1] 1996 disk trend report–rigid disk drives, figure 2–unit shipment summary. http://www.disktrend.com.

[2] Berkeley web trace. http://tracehost.cs.berkeley.edu/web/, 1998.

[3] Seagate unveils hefty, fast cheetah drives, March 2001.

[4] The distributed-parallel storage system (dpss) home pages. http://www-didc.lbl.gov/DPSS/, June 2004.

[5] Hitachi introduces 1-terabyte hard drive. http://www.pcworld.com/article/128400/hitachi_introduces_1terabyte_hard_drive.html, January 2007.

[6] Umass trace repository. http://traces.cs.umass.edu/index.php/Storage/Storage, December 2009.

[7] Japans k computer tops 10 petaflop/s to stay atop top500 list, November 2011.

[8] Seagate is the first manufacturer to break the capacity ceiling with a new 4tb goflex desk drive. http://www.seagate.com/ww/v/index.jsp?locale=en-US&name=goflex-desk-4tb-capacity-seagate-pr&vgnextoid=e07c2d857df32310VgnVCM1000001a48090aRCRD, September 2011.

[9] Top 500 supercomputer sites, March 2012.

[10] Robert B. Abernethy. *The New Weibull Handbook 5th edition.* Barringer & Associates, Inc, Humble, TX, USA, 2010.

[11] Khalil S. Amiri. Scalable and manageable storage systems. Carnegie Mellon University, December 2000.

[12] K. Bellam, A. Manzanares, X. Ruan, X. Qin, and Y.-M. Yang. Improving reliability and energy efficiency of disk systems via utilization control. In *Proc. IEEE Symp. Computers and Comm.*, 2008.

[13] Soren Bergmann and Steffen Strassburger. Challenges for the automatic generation of simulation models for production systems. In *2010 Summer Simulation Multiconference*, SummerSim '10, pages 545–549, San Diego, CA, USA, 2010. Society for Computer Simulation International.

[14] M. Blaum, J. Brady, J. Bruck, and J. Menon. Evenodd: an optimal scheme for tolerating double disk failures in raid architectures. In *Computer Architecture, 1994., Proceedings the 21st Annual International Symposium on*, pages 245 – 254, apr 1994.

[15] Francieli Zanon Boito, Rodrigo Virote Kassick, and Philippe O. A. Navaux. The impact of applications' i/o strategies on the performance of the lustre parallel file system. *Int. J. High Perform. Syst. Archit.*, 3:122–136, May 2011.

[16] R.E Brown and J.R. Ochoa. Distribution system reliability: default data and model validation. In *IEEE Transactions on Power Systems*, pages 704–709, March 1998.

[17] W.A. Burkhard and J. Menon. Disk array storage system reliability. In *Proc. 23rd Int'l Symp. Fault-Tolerant Comp.*, pages 432–441, 1993.

[18] Philip H. Carns, Bradley W. Settlemyer, and Walter B. Ligon, III. Using server-to-server communication in parallel file systems to simplify consistency and improve performance. In *Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, SC '08, pages 6:1–6:8, Piscataway, NJ, USA, 2008. IEEE Press.

[19] D. Colarelli and D. Grunwald. Massive arrays of idle disks for storage archives. In *Proc. ACM/IEEE Conf. Supercomputing*, pages 1–11, 2002.

[20] Gerry Cole. Estimating drive reliability in desktop computers and consumer electronics systems. Seagate Personal Storage Group, 2000.

[21] Bryan Dodson. *Weibull Analysis*. ASQC Quality Press, Milwaukee, WI,USA, 1994.

[22] F. Douglis, P. Krishnan, and B. Marsh. Thwarting the power-hungry disk. In *Proc. USENIX Winter 1994 Technical Conf.*, pages 23–23, 1994.

[23] F. Douglis, P. Krishnan, and B. Marsh. Thwarting the power-hungry disk. In *Proc. USENIX Winter 1994 Technical Conf.*, pages 23–23, 1994.

[24] B. Eckart, Xin Chen, Xubin He, and S.L. Scott. Failure prediction models for proactive fault tolerance within storage systems. In *Modeling, Analysis and Simulation of Computers and Telecommunication Systems, 2008. MASCOTS 2008. IEEE International Symposium on*, pages 1 –8, sept. 2008.

[25] J.G. Elerath. Specifying reliability in the disk drive industry: No more mtbf's. pages 194–199, 2000.

[26] J.G. Elerath and M. Pecht. Enhanced reliability modeling of raid storage systems. In *Proc. IEEE/IFIP Int'l Conf. Dependable Sys. and Networks*, 2007.

[27] Hyeonsang Eom and Jeffrey K. Hollingsworth. Speed vs. accuracy in simulation for i/o-intensive applications. In *IPDPS*, pages 315–322. IEEE Computer Society Press, 2000.

[28] Blake G. Fitch, Aleksandr Rayshubskiy, Michael C. Pitman, T. J. Christopher Ward, and Robert S. Germain. Using the active storage fabrics model to address petascale storage challenges. In *Proceedings of the 4th Annual Workshop on Petascale Data Storage*, PDSW '09, pages 47–54, New York, NY, USA, 2009. ACM.

[29] Richard Freitas, Joseph Slember, Wayne Sawdon, and Chiu Lawrence. Gpfs scans 10 billion files in 43 minutes. San Jose, CA, USA, 2011. IBM Advanced Storage Laboratory.

[30] E. Grochowski and R.F. Hoyt. Future trends in hard disk drives. *Magnetics, IEEE Transactions on*, 32(3):1850 –1854, may 1996.

[31] Jorge Guerra, Himabindu Pucha, Joseph Glider, Wendy Belluomini, and Raju Rangaswami. Cost effective storage using extent based dynamic tiering. In *Proceedings of the 9th USENIX conference on File and stroage technologies*, FAST'11, pages 20–20, Berkeley, CA, USA, 2011. USENIX Association.

[32] S. Gurumurthi, A. Sivasubramaniam, M. Kandemir, and H. Franke. Drpm: dynamic speed control for power management in server class disks. In *Proc. Int'l Symp. Computer Architecture*, pages 169–179, June 2003.

[33] Ibrahim F. Haddad. Pvfs: A parallel virtual file system for linux clusters. *Linux J.*, 2000, November 2000.

[34] D.P. Helmbold, D.E. Long, T.L. Sconyers, and B. Sherrod. Adaptive disk spin—down for mobile computers. *Mob. Netw. Appl.*, 5(4):285–297, 2000.

[35] G.F. Hughes and J.F. Murray. Reliability and security of raid storage systems and d2d archives using sata disk drives. *ACM Trans. Storage*, 1(1):95–107, Dec. 2004.

[36] G.F. Hughes, J.F. Murray, K. Kreutz-Delgado, and C. Elkan. Improved disk-drive failure warnings. *Reliability, IEEE Transactions on*, 51(3):350 – 357, sep 2002.

[37] Maximum Institution Inc. 2002.

[38] S. Jin and A. Bestavros. Gismo: A generator of internet streaming media objects and workloads. *ACM SIGMETRICS Performance Evaluation Review*, November 2001.

[39] Hawkins John and Bod00E9n Mikael. The applicability of recurrent neural networks for biological sequence analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2:243–253, 2005.

[40] Steven W. Schlosser John S. Bucy, Jiri Schindler and Gregory R. Ganger. The disksim simulation environment version 4.0 reference manual. 2008.

[41] Mahmut Kandemir, Seung Woo Son, and Mustafa Karakoy. Improving disk reuse for reducing power consumption. In *Proceedings of the 2007 international symposium on Low power electronics and design*, ISLPED '07, pages 129–134, New York, NY, USA, 2007. ACM.

[42] P Krishnan, M P Long, and Scott J Vitter. Adaptive disk spindown via optimal rent-to-buy in probabilistic environments. Technical report, Durham, NC, USA, 1995.

[43] Samuel Lang, Philip Carns, Robert Latham, Robert Ross, Kevin Harms, and William Allcock. I/o performance challenges at leadership scale. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, SC '09, pages 40:1–40:12, New York, NY, USA, 2009. ACM.

[44] Chunhua Li, Ke Zhou, and Dan Feng. Capturing the object behavior for storage system evaluation. *Int. J. High Perform. Comput. Netw.*, 6:226–233, December 2010.

[45] Dong Li and Jun Wang. Eeraid: energy efficient redundant and inexpensive disk array. In *Proceedings of the 11th workshop on ACM SIGOPS European workshop*, EW 11, New York, NY, USA, 2004. ACM.

[46] K. Li, R. Kumpf, P. Horton, and T. Anderson. A quantitative analysis of disk drive power management in portable computers. In *Proc. USENIX Winter Technical Conf.*, pages 22–22, 1994.

[47] Seagate Technology LLC. Product manual: Cheetak 15k.6 sas. Scotts, CA, USA, September 2008. Seagate Technology LLC.

[48] Tim Lynam, John Drewry, Will Higham, and Carl Mitchell. Adaptive modelling for adaptive water quality management in the great barrier reef region, australia. *Environ. Model. Softw.*, 25:1291–1301, November 2010.

[49] Adam Manzanares, Xiaojun Ruan, Shu Yin, and Mais Nijim. Energy-aware prefetching for parallel disk systems: Algorithms, models, and evaluation. *IEEE Int'l Symp. on Network Computing and Applications*, 2009.

[50] Adam C. Manzanares. Energy efficient pre-fetching–models to implementation. Auburn University, April 2010.

[51] C. Mee and Eric Daniel. *Magnetic Storage Handbook*. McGraw-Hill, Inc., New York, NY, USA, 2 edition, 1996.

[52] J. Menon. A performance comparison of raid-5 and log-structured arrays. In *High Performance Distributed Computing, 1995., Proceedings of the Fourth IEEE International Symposium on*, pages 167 –178, aug 1995.

[53] David Nagle, Denis Serenyi, and Abbie Matthews. The panasas activescale storage cluster: Delivering scalable high bandwidth storage. In *Proceedings of the 2004 ACM/IEEE conference on Supercomputing*, SC '04, pages 53–, Washington, DC, USA, 2004. IEEE Computer Society.

[54] Athanasios E. Papathanasiou and Michael L. Scott. Power-efficient server-class performance from arrays of laptop disks. 2004.

[55] J.-F. Pâris, T.J. Schwarz, and D.D.E. Long. Evaluating the reliability of storage systems. In *Proc. IEEE Int'l Symp. Reliable and Distr. Sys.*, 2006.

[56] David A. Patterson, Garth Gibson, and Randy H. Katz. A case for redundant arrays of inexpensive disks (raid). In *SIGMOD '88: Proceedings of the 1988 ACM SIGMOD international conference on Management of data*, pages 109–116, New York, NY, USA, 1988. ACM.

[57] Juan Piernas, Jarek Nieplocha, and Evan J. Felix. Evaluation of active storage strategies for the lustre parallel file system. In *Proceedings of the 2007 ACM/IEEE conference on Supercomputing*, SC '07, pages 28:1–28:10, New York, NY, USA, 2007. ACM.

[58] E. Pinheiro and R. Bianchini. Energy conservation techniques for disk array-based servers. In *Proc. 18th Int'l Conf. Supercomputing*, 2004.

[59] E. Pinheiro, R. Bianchini, E. Carrera, and T. Heath. Load balancing and unbalancing for power and performance in cluster-based systems. *Proc. Workshop Compilers and Operating Sys. for Low Power*, September 2001.

[60] E. Pinheiro, R. Bianchini, and C. Dubnicki. Exploiting redundancy to conserve energy in storage systems. In *Proc. Joint Int'l Conf. Measurement and Modeling of Computer Systems*, 2006.

[61] E. Pinheiro, W.-D. Weber, and L.A. Barroso. Failure trends in a large disk drive population. In *Proc. USENIX Conf. File and Storage Tech.*, February 2007.

[62] Eduardo Pinheiro, Ricardo Bianchini, and Cezary Dubnicki. Exploiting redundancy to conserve energy in storage systems. *SIGMETRICS Perform. Eval. Rev.*, 34(1):15–26, 2006.

[63] A. Polze, P. Troandger, and F. Salfner. Timely virtual machine migration for pro-active fault tolerance. In *Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW), 2011 14th IEEE International Symposium on*, pages 234 –243, march 2011.

[64] Drew Roselli, Jacob R. Lorch, and Thomas E. Anderson. A comparison of file system workloads. In *ATEC '00: Proceedings of the annual conference on USENIX Annual Technical Conference*, pages 4–4, Berkeley, CA, USA, 2000. USENIX Association.

[65] X. J. Ruan, A. Manzanares, K. Bellam, Z. L. Zong, and X. Qin. Daraw: A new write buffer to improve parallel I/O energy-efficiency. In *Proc. ACM Symp. Applied Computing*, 2009.

[66] X.-J. Ruan Run, A. Manzanares, S. Yin, Z.-L. Zong, and X. Qin. Performance evaluation of energy-efficient parallel I/O systems with write buffer disks. In *Proc. 38th Int'l Conf. Parallel Processing*, Sept. 2009.

[67] Robert G. Sargent. Verification and validation of simulation models. In *WSC '05: Proceedings of the 37th conference on Winter simulation*, pages 130–143. Winter Simulation Conference, 2005.

[68] Robert G. Sargent. Verification and validation of simulation models. In *Proceedings of the 37th conference on Winter simulation*, WSC '05, pages 130–143. Winter Simulation Conference, 2005.

[69] K. Bernhard Schiefer and Gary Valentin. Db2 universal database performance tuning. *IEEE Data Eng. Bull.*, 22(2):12–19, 1999.

[70] S. Schlesinger, RE. Crosbie, RE Gagne, and Innie GSd. Terminology for model credibility. In *Simulation 32*, pages 103–104, 1979.

[71] B. Schroeder and G.A. Gibson. Disk failures in the real world: what does an mttf of 1,000,000 hours mean to you? In *Proc. USENIX Conf. File and Storage Tech.*, page 1, 2007.

[72] S. Shah and J.G. Elerath. Reliability analysis of disk drive failure mechanisms. In *Proc. Annual Reliability and Maintainability Symp.*, pages 226–231, 2005.

[73] H. Shen, Mohan Kumar, S.K. Das, and Z. Wang. Energy-efficient caching and prefetching with data consistency in mobile distributed systems. In *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, page 67, april 2004.

[74] Sean M. Snyder, Shimin Chen, Panos K. Chrysanthis, and Alexandros Labrinidis. Qmd: exploiting flash for energy efficient disk arrays. In *Proceedings of the Seventh International Workshop on Data Management on New Hardware*, DaMoN '11, pages 41–49, New York, NY, USA, 2011. ACM.

[75] S. W. Son, M. Kandemir, and A. Choudhary. Software-directed disk power management for scientific applications. In *Proc. IEEE Int'l Parallel and Distr. Processing Symp.*, 2005.

[76] S.W. Son and M. Kandemir. Energy-aware data prefetching for multi-speed disks. In *Proc. Int'l Conf. Comp. Frontiers*, 2006.

[77] Huaiming Song, Yanlong Yin, Xian-He Sun, Rajeev Thakur, and Samuel Lang. Server-side i/o coordination for parallel file systems. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '11, pages 17:1–17:11, New York, NY, USA, 2011. ACM.

[78] IDEMA Standards. Specification of hard disk drive reliability. pages Document Number R2–98.

[79] Jan Stender, Björn Kolbeck, Felix Hupfeld, Eugenio Cesario, Erich Focht, Matthias Hess, Jesús Malo, and Jonathan Martí. Striping without sacrifices: maintaining posix semantics in a parallel file system. In *First USENIX Workshop on Large-Scale Computing*, pages 6:1–6:8, Berkeley, CA, USA, 2008. USENIX Association.

[80] A. Thomasian and M. Blaum. Mirrored disk organization reliability analysis. *IEEE Trans. Computers*, 55(12):1640–1644, 2006.

[81] P.J. Varman and R.M. Verma. Tight bounds for prefetching and buffer management algorithms for parallel I/O systems. *IEEE Trans. Parallel Distrib. Syst.*, 10(12):1262–1275, 1999.

[82] J. Wang, H.-J. Zhu, and D. Li. eraid: Conserving energy in conventional disk-based raid system. *IEEE Trans. Computers*, 57(3):359–374, 2008.

[83] Jun Wang, Xiaoyu Yao, and Huijun Zhu. Exploiting in-memory and on-disk redundancy to conserve energy in storage systems. *IEEE Trans. Comput.*, 57:733–747, June 2008.

[84] Jun Wang, Huijun Zhu, and Dong Li. eraid: Conserving energy in conventional disk-based raid system. *IEEE Transactions on Computers*, 57(3):359–374, 2008.

[85] Charles Weddle, Mathew Oldham, Jin Qian, An-I Andy Wang, Peter Reiher, and Geoff Kuenning. Paraid: a gear-shifting power-aware raid. In *FAST '07: Proceedings of the 5th USENIX conference on File and Storage Technologies*, pages 30–30, Berkeley, CA, USA, 2007. USENIX Association.

[86] Sage A. Weil, Scott A. Brandt, Ethan L. Miller, Darrell D. E. Long, and Carlos Maltzahn. Ceph: a scalable, high-performance distributed file system. In *Proceedings of the 7th symposium on Operating systems design and implementation*, OSDI '06, pages 307–320, Berkeley, CA, USA, 2006. USENIX Association.

[87] Andreas Weissel, Björn Beutel, and Frank Bellosa. Cooperative I/O: a novel I/O semantics for energy-aware applications. In *Proc. the 5th Symp. Operating Systems Design and Implementation*, pages 117–129, New York, NY, USA, 2002. ACM.

[88] Brent Welch, Marc Unangst, Zainul Abbasi, Garth Gibson, Brian Mueller, Jason Small, Jim Zelenka, and Bin Zhou. Scalable performance of the panasas parallel file system. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies*, FAST'08, pages 2:1–2:17, Berkeley, CA, USA, 2008. USENIX Association.

[89] T. Xie. Sea: A striping-based energy-aware strategy for data placement in raid-structured storage systems. *IEEE Trans. Computers*, 57(6):748–761, June 2008.

[90] T. Xie and Y. Sun. Sacrificing reliability for energy saving: Is it worthwhile for disk arrays? In *Proc. IEEE Symp. Parallel and Distr. Processing*, pages 1–12, April 2008.

[91] Tao Xie. Sea: A striping-based energy-aware strategy for data placement in raid-structured storage systems. *IEEE Transactions on Computers*, 57:748–761, 2008.

[92] Tao Xie and Hui Wang. Micro: A multilevel caching-based reconstruction optimization for mobile storage systems. *Computers, IEEE Transactions on*, 57(10):1386 –1398, oct. 2008.

[93] Q. Xin, J.E. Thomas, S.J. Schwarz, and E.L. Miller. Disk infant mortality in large storage systems. In *Proc. IEEE Int'l Symp. Modeling, Analysis, and Simulation of Computer and Telecomm. Sys.*, 2005.

[94] Qin Xin, E.L. Miller, and S.J.T.J.E. Schwarz. Evaluation of distributed recovery in large-scale storage systems. In *High performance Distributed Computing, 2004. Proceedings. 13th IEEE International Symposium on*, pages 172 – 181, june 2004.

[95] Qin Xin, E.L. Miller, T. Schwarz, D.D.E. Long, S.A. Brandt, and W. Litwin. Reliability mechanisms for very large storage systems. In *Mass Storage Systems and Technologies, 2003. (MSST 2003). Proceedings. 20th IEEE/11th NASA Goddard Conference on*, pages 146 – 156, april 2003.

[96] Ying Xu and Brett D. Fleisch. Nfs-cc: tuning nfs for concurrent read sharing. *Int. J. High Perform. Comput. Netw.*, 1:203–213, December 2004.

[97] J. Yang and F.-B. Sun. A comprehensive review of hard-disk drive reliability. In *Proc. Annual Reliability and Maintainability Symp.*, 1999.

[98] Q. Yang and Y.-M. Hu. DCD - Disk Caching Disk: A new approach for boosting I/O performance. In *Proc. Int'l Symp. Computer Architecture*, pages 169–169, May 1996.

[99] S. Yin, X. Ruan, A. Manzanares, and X. Qin. How reliable are parallel disk systems when energy-saving schemes are involved? In *Proc. IEEE International Conference on Cluster Computing (CLUSTER)*, 2009.

[100] John Zedlewski, Sumeet Sobti, Nitin Garg, Fengzhou Zheng, Arvind Krishnamurthy, and Randolph Wang. Modeling hard-disk power consumption. In *Proceedings of the 2nd USENIX Conference on File and Storage Technologies*, pages 217–230, Berkeley, CA, USA, 2003. USENIX Association.

[101] Junyao Zhang, Pengju Shang, and Jun Wang. A scalable reverse lookup scheme using group-based shifted declustering layout. In *Parallel Distributed Processing Symposium (IPDPS), 2011 IEEE International*, pages 604 –615, may 2011.

[102] Q.-B. Zhu, F.M. David, C.F. Devaraj, Z.-M. Li, Y.-Y. Zhou, and P. Cao. Reducing energy consumption of disk storage using power-aware cache management. In *Proc. Int'l Symp. High Performance Comp. Arch.*, page 118, Washington, DC, USA, 2004.

[103] Qingbo Zhu, Zhifeng Chen, Lin Tan, Yuanyuan Zhou, Kimberly Keeton, and John Wilkes. Hibernator: helping disk arrays sleep through the winter. *SIGOPS Oper. Syst. Rev.*, 39:177–190, October 2005.