Reliability Issues In MEMS-Based Storage Systems[†]

Bo Hong, Thomas Schwarz, S.J., Scott A. Brandt and Darrell D. E. Long

Storage Systems Research Center
University of California Santa Cruz

Abstract

Abstract goes here.

1 Introduction

Magnetic disks have dominated secondary storage for decades. A new class of secondary storage devices based on microelectromechanical systems (MEMS) is a promising non-volatile secondary storage technology currently being developed [5, 24, 34, 35]. With fundamentally different underlying architectures, MEMS-based storage promises seek times 10-20 times faster than hard disks, storage densities 10 times greater, and power consumption one to two orders of magnitude lower. It can provide several to tens of gigabyte non-volatile storage in a single chip as small as a quarter, with low entry cost, low shock sensitivity, and potentially high embedded computing power [5, 15, 28]. For all of these reasons, MEMS-based storage is an appealing next-generation storage technology.

MEMS-based storage devices are also expected to be more reliable than hard disks thanks to their architectures, miniature structures, and manufacture processes [6, 15, 28]. Unlike disks, there is no rotating mechanical components in MEMS devices and hundreds to thousands read/write probe tips can access data simultaneously. Thus MEMS devices has low shock sensitivity and can tolerate a number of tip failures if designed carefully. Likes disk, MEMS suffers random bit errors, off-track interference, media defects, and probe tip failures.

Schlosser *et al*. [28] proposed to employ system-level RAID models among different tip storage locations to protect against local tip failures and media defects. However, we find that such a straightforward approach is unlikely to achieve good performance and space efficiency; Instead, using Error Correction Coding (ECC) would yield better reliability with lower space overheads. We also propose a probe tip grouping scheme to minimize the impact of off-track interference.

Due to their limited storage capacity, storage systems built on MEMS-based storage require one to two orders of magnitude more MEMS devices, also the corresponding connection components, than disks to meet their capacity requirements. These can significantly undermine system reliability and increase system costs. Nevertheless, thanks to their higher bandwidth as well as limited capacity, MEMS devices typically have much shorter times than disks during data reconstruction after device failures, which can significantly reduce the risk of data loss. The low entry cost of MEMS devices also makes it more flexible to add on-line spares in storage systems to improve their reliability.

We propose to integrate multiple MEMS devices, organized as RAID-5, into a *MEMS storage enclosure*. MEMS storage enclosures are the building block of MEMS-based storage systems, whose role is exactly the same as disks' — providing reliable persistent storage. There are several on-line spares in a MEMS enclosure. The enclosure notifies the host system, the maintenance personnel and/or the end users when it runs out of spares. The enclosure can be either upgraded by a new enclosure or replenished with new spares to increase its life time, depending upon users' decisions. We find that a preventive replacement/repair strategy can make MEMS enclosure highly reliable with the cost of moderately increased maintenance frequencies. We also provide an equation to evaluate the economy of maintaining on-line spares in large storage servers based on MEMS storage.

2 MEMS-based Storage

It is important to note that because MEMS-based storage devices are still in their infancy, many of the details are still uncertain. There are several proposed architectures [5, 6, 24, 34, 35], and we have based the physical

[†]This research is supported by the National Science Foundation under grant number CCR-073509 and the Institute for Scientific Computation Research at Lawrence Livermore National Laboratory under grant number SC-20010378.

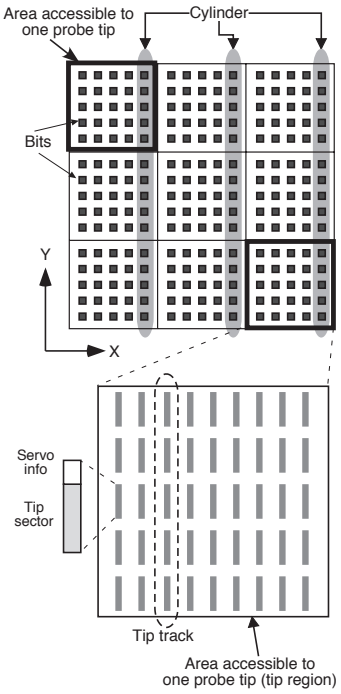


Figure 1: Data layout on a MEMS device.

Table 1: Default MEMS-based storage device parameters.

Per-sled capacity	3.2 GB
Max. throughput	76 MB/s
Number of tips	6400
Maximum concurrent tips	1280

parameters of our experimental model on the specification from Carnegie Mellon University (CMU) [14, 28]. While the exact reliability numbers depend upon the details of that specification, the techniques themselves do not.

A MEMS-based storage device is comprised of two main components: groups of probe tips called *tip arrays* that are used to access data on a movable *media sled*. In a modern disk drive, data is accessed by means of an arm that seeks in one dimension above a rotating platter. In a MEMS device, the entire media sled is positioned in the x and y directions by electrostatic forces while the heads remain stationary.[1] Another major difference between a MEMS-based storage device and a disk is that on a MEMS device, multiple tips can be active at the same time. Data can be then be striped across multiple tips, allowing a considerable amount of parallelism. However, power and heat considerations limit the number of probe tips that can be active simultaneously; it is estimated that 200 to 2000 probes will actually be active at once.

Figure 1 illustrates the low level data layout of a MEMS-based storage device. The media sled is logically broken into *tip regions*, defined by the area that is accessible by a single head, approximately 2500 by 2500 bits in size. Each tip in the MEMS device can only read the data in its own tip region; this limits the maximum sled movement to the dimensions of a single tip region. The smallest data access unit in a MEMS-based storage device is a *logical sector*, which is also the basic Error Correction Coding (ECC) unit. A logical sector is striped across multiple simultaneously-accessible tips, each of which reads/writes a *tip stripe*. Each tip stripe, identified by the tuple $\langle x, y, tip \rangle$, has its own servo information for positioning. The set of bits accessible to a single tip with the same x coordinate is called a *tip track*, and the set of all bits (under all tips) with the same x coordinate is referred to as a *cylinder*. For faster access, logical blocks can be striped across logical sectors. Table 1 summarizes the parameters of the CMU **G2** MEMS model [28].

3 Related Work

Although MEMS-based storage is still in its infancy and no public literature is available on its reliability, the MEMS technology itself has played important roles in automotive industries, medical technologies, communications, and so on [1]. Among them, Digital Micromirror Devices (DMD) is a commercial MEMS-based digital imaging technology developed by Texas Instruments (TI). Douglass [9, 10] reported and estimated its Mean Time Between Failure (*MTBF*) was 650,000 hours.

Scientists and engineers in disk manufacturers have long put efforts on improving the reliability of disks through various techniques so that data stored on these devices can be retrieved back correctly later with extremely high probabilities. Meanwhile, storage system designers are have built more reliable systems, coping with imperfect storage devices and other system components and environmental factors.

Should mention ECC coding in disks here.

RAID (Redundant Arrays of Independent Disks) [7, 13, 25] have been used for many years to improve both system reliability and performance. Traditionally, system designers were more concerned with system performance during recovery than with reliability. Menon and Mattson and Thomasian [21, 22, 33] evaluated the performance of dedicated sparing [11], distributed sparing [22], and parity sparing [27] under the normal and data recovery modes of RAID systems. Muntz and Lui [23] proposed that a disk

[1] Some MEMS storage device designs, like the IBM Millipede, fix the sled and move the heads. The effect is the same—the heads move relative to the media.

array of n disks be declustered by grouping the blocks in the disk array into reliability sets of size g and analyzed its performance under failure recovery.

Disk manufacturers are widely using S.M.A.R.T (Self-Monitoring Analysis and Reporting Technology) to recognize conditions that indicate a drive failure and provide sufficient warning before an actual failure occurs [2, 30, 18]. The preventive replacement strategy used in our MEMS storage enclosures can be viewed as a coarse-grained device failure predictor.

4 Internal Designs for MEMS-based Storage Reliability

Besides density, performance, and capacity, reliability is always a demanding concern in the secondary storage industry. The Mean Time To Failure (*MTTF*) of state-of-the-art high end SCSI disk drives are claimed to be more than one million (10^6) hours with nonrecoverable read error rates of one out of 10^{15} sectors [31]. Even cheap commodity IDE disk drives are claimed to have *MTTF* of more than 10^5 hours with nonrecoverable read error rates of one out of 10^{14} bits [31].

Like hard disks, MEMS-based storage devices also suffer from shared component failures. For instance, the failures of actuators, springs, internal power wires, and on-board circuits can result in all data on the device inaccessible. Such failures can be tolerable with the help of inter-device redundancy, which will be discussed in Section 5. Besides these "hard" failures, random bit errors, probe tip failures, media defects, and off-track interference can result in retrieving incomplete or incorrect data. Intra-device schemes, such as ECC codes, internal RAID organizations, spare components, and specialized data layouts, can be used to alleviate these "soft" effects and make the device more reliable.

4.1 Off-Track Interference

MEMS-based storage devices write and retrieve data through a group of simultaneously active probe tips. Off-track interference occurs if the MEMS media sled is not positioned to the right place during writes. It makes the data being written hard to be retrieved later; even worse, it can destroy data in adjacent bit columns. In the worse case, the amount of data being destroyed is the same as the amount of data being written.

If the destroyed data belongs to one or few logical sectors (the minimum data access unit in a MEMS device), it is unlikely to recover it by the means of intra-device redundancy because a logical sector is also the basic encoded data reliability unit, which is not supposed to tolerate such a level of bursts of errors due to the consideration of space efficiency. However, if the group of simultaneously active read/write tips are organized carefully, the amount of destroyed data in one logical sector can be limited and data can be recovered through intra-device redundancy.

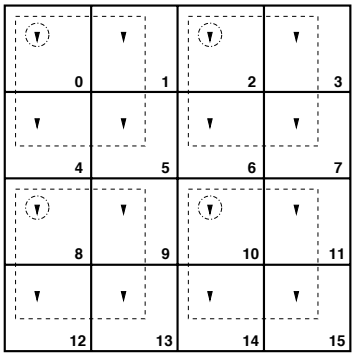


Figure 2: A MEMS tip grouping scheme that can minimize the effects of off-track interference. In this example, a data sector is accessed by four tips. The four tips in the same rectangle access data sectors on the even-numbered bit columns; the four tips labeled by circles (0, 2, 8, and 10) access data sectors on the odd-numbered bit columns.

We assume that the media sled mis-positioning during writes will be no larger than one bit column, *i.e.* writing even-numbered bit columns can only affect their adjacent odd-numbered bit columns, and vice versa. Thus if a tip group for data sectors on even-numbered bit columns has at most one overlapped tip with any of the tip groups for data sectors on odd-numbered bit columns, off-track interference caused by this group will only affect a small fraction of data in one logical sector.

Suppose there are $80 \times 80 = 6,400$ tips and a logical sector of 512 bytes is accessed by a group of 64 tips. The amount of data each tip accesses is more than eight bytes because of extra servo information bits and ECC bits. For logical sectors on even-numbered bit columns, we can easily configure the 6,400 tips into 100 groups so that any two tips within one group are apart from each other by at least one tip, *i.e.* any two tips are apart from each other by at least one one tip region. For logical sectors on odd-numbered bit columns, a group of tips consists of 64 tips from different tip groups (totally 100) partitioned for logical sectors on even-numbered bit columns. Figure 2 shows an example of such tip grouping. In this way, any off-track interference can affect only one tip stripe within one logical sector. Another benefit of this tip grouping scheme is that media defects with diameters less than one

tip region can only affect one tip stripe within one logical sector. In short, grouping tips in this way limits the scale of bursts of errors, which facilitates the design of efficient ECC coding for MEMS-based storage devices.

4.2 Random Bit Errors — ECC Codeword Sizes

One of the most basic requirements on secondary storage devices is that data written to these devices should be later retrieved correctly with extremely high probabilities. However, random bit errors occur all the time due to the nature of magnetic recording media, signal noises, and even alpha particle bombardment. Making "perfect" or "near perfect" devices is either prohibitively expensive and difficult, or inefficient in bit densities, or both.

Typically, error correction coding (ECC) is used to mask random bit errors in secondary storage devices. By adding a few bytes of redundant data, ECC can detect and/or correct certain numbers of bit errors and guarantee that the reliability of data achieves system requirements.

There are a rich body of research on error correction coding theory. We choose Reed-Solomon (RS) codes [17] in our further analysis. The RS codes are maximum distance separable (MDS) codes. A Reed-Solomon code is specified as RS(n,k) with s-bit symbols, which means that the encoder takes k data symbols of s bits each and adds parity symbols to make an n symbol codeword. Given a symbol size of s, the maximum codeword length n for a RS code is $2^s - 1$. A RS(n,k) code can correct up to $\frac{n-k}{2}$ symbols that contain errors in a codeword. Here we assume that there is enough embedded computing power in a MEMS device to run such advanced codes.

Given the random bit error rate and the same coding technique, longer codewords generally achieve the desired reliability level with higher space efficiency than shorter codewords. In our context, a codeword is a basic reliability unit. We do some simple calculations to show the space efficiency under different data unit lengths, 512 bytes and 8 bytes, respectively.

The raw bit error rates of perpendicular magnetic recording are expected to be in the order of 10^{-6} [8]. We conservatively assumes that such rates range from 10^{-4} to 10^{-7}. Table 2 shows the relations among reliability unit sizes, data failure rates, and space utilizations under different assumptions on the raw bit error rate. Note that the symbol size is 9 bits when a reliability unit contains 512-byte user data. For instance, under the random bit error rate of 10^{-4} the probability of unsuccessfully reading a reliability unit containing 512-byte user data and $2 \times 13 = 26$ parity symbols is 1.1×10^{-16}. Table 2 clearly shows that longer codewords result in much better space efficiency.

A tip sector, including 8-byte user data, necessary ECC bits, and servo information, was claimed to be the minimum data access unit in MEMS [15, 14, 28]. It implies that a tip sector is the basic encoding unit or reliability unit. Since the size of a tip sector is chosen to be small for the purpose of data transfer parallelism, ECC coding on such small data units is very inefficient, about 40–67% as shown in Table 2. Moreover, redundancy on the tip sector level cannot protect again tip failures and media defects. Thus we believe that it is unlikely for a tip sector to be the minimum data access unit in MEMS and redundancy should be done on larger data units.

4.3 Burst Bit Erasures — Internal RAID or Longer ECC Codewords?

With thousands of tips and its miniature structure, it is probable for a MEMS-based storage device to face probe tip crashes and media defects in its lifetime. Such failures result in a set of contiguous bits irretrievable and usually the device can notice where the irretrievable bits are. So tip crashes and media defects cause data "erasures" instead of data "errors". The probability of tip crashes and media defects in a short period is much less than the probability of random bit errors. Moreover, tip failures and media defects are typically local, which are expected to be within 1–4 tips and tip regions [15]. Thus it makes sense to protect against random bit errors at the most basic reliability unit and against infrequent tip failures and media defects among multiple basic reliability units.

Probe tip failures and media defects are essentially equivalent, both of which cause an region of bits inaccessible. A small fraction of spare tips thus tip regions, say 1%, can be reserved to hold recovered data.

4.3.1 Internal RAID

The basic idea of internal RAID in MEMS devices [28] is to use the system-level RAID model in the internal MEMS data layout. User data is striped across a group of tips and several extra parity tips (so parity data) are employed against failures of data tips. Unfortunately, the authors did not further their exploration in this topic.

Simply using the RAID model within MEMS cannot achieve good performance and space efficiency together. When using RAID in the system level, data is assumed to be correct if it can be retrieved from a device. However, such an assumption does not hold within a device, as discussed in Section 4.2. Thus in our context each data stripe

Table 2: Data reliability and space efficiency under different random bit error rates.

Random Bit Error Rate	Reliability Unit Size (Byte)					
	512			8		
	Failure Rate	Tolerable Symbol Errors	Space Efficiency	Failure Rate	Tolerable Symbol Errors	Space Efficiency
10^{-4}	1.1×10^{-16}	13	94.4%	1.1×10^{-16}	6	40%
10^{-5}	2.2×10^{-16}	7	96.8%	$\sim 10^{-17}$	4	50%
10^{-6}	$\sim 10^{-17}$	5	97.7%	$\sim 10^{-17}$	3	57.1%
10^{-7}	$\sim 10^{-17}$	4	98.1%	1.1×10^{-16}	2	66.7%

unit should also be a basic reliability unit if we simply apply the system-level RAID model in MEMS. To achieve higher space efficiency, larger stripe unit sizes are preferred, which on the contrary increases data transfer times and decreases throughputs. Even with smaller data stripe units, say 8 bytes, data transfer times can be doubled compared to the raw MEMS performance because of the ECC overhead, as shown in Table 2.

One variant of the internal RAID approach is that striping the encoded user data and its parity bits across a group to data tips and several parity tips, respectively. Encoding the whole user data unit instead of each tip stripe can achieve good reliability as well as space efficiency. Because of the data tip grouping scheme described in Section 4.1 and the expected number of tip crashes, no more than two tips within one tip group are expected to crash simultaneously. Thus, computationally simpler parity coding schemes, such as EVENODD coding [4], can be used here. Because parity data is relatively smaller than user data, inefficient storage usage due to the reliability guarantee for parity data is acceptable. During data recovery, the decoder uses the remaining unreliable user data and parity data (probably unreliable too) to reconstruct the lost data as much as possible. And then the internal codes of the user and parity data reliability units are further used to correct remaining errors. This process can be iterated more than once until the resulting data looks correct.

This internal RAID variant can be extended across multiple encoded user data units and achieve better space efficiency. In general, the amount of user data in a parity group should be no larger than the minimum data access size of high-level file systems to avoid performance penalties during data updates. Otherwise, every data update requires reading the old parity data first and then writes new user and parity data concurrently. Such operations double the transfer time and also require necessary media sled turnaround times. In other words, partial stripe updates, in the RAID term, hurt performance and should be avoided. The constraint essentially limits the amount of user data that can be grouped together.

However, if the on-board memory of a MEMS device is large enough to hold all parity data of the whole device, the performance problem of reading old parity data can be avoided. In this scheme, data of the whole device are parity-checked by four parity tips and protected against failures up to four tips or tip regions. Because all parity data is cached, there is no extra overhead for partial stripe updates. Under the CMU **G2** model with 6,400 tips and 3.2 GB storage as described in Section 2, each tip region stores 0.5 MB data. Considering the possible number of simultaneously failed tips and space overheads for the reliability guarantee of parity data, roughly 3.5–4 MB on-board memory is needed.

Although parity-checking the whole device can achieve better space efficiency, there are several disadvantages: the space saving is not significant, roughly 3%. The cost of several megabyte on-board memory may exceed the benefit of saved storage. Complicated Reed-Solomon codes, instead of simple EVENODD codes, have to be used to recover from up to four concurrent tip crashes. Because there are thousands tips in a MEMS device, parity-checking the whole device implies that the RS codeword contains thousands of symbols. This further increases the computational cost of RS codes and the size of data structures required for RS coding computations.

4.3.2 Longer ECC Codewords

Burst bit erasures can be corrected by adding more parity bits and using longer ECC codewords. For instance, we assume that the data unit size is 512 bytes and the encoded data is striped across 64 probe tips. Following the tip grouping scheme described in Section 4.1 and considering the expected number of tip crashes, no more than two tips within one tip group are expected to crash concurrently. Therefore, another two tip stripe worth of parity bits can recover the lost data due to crashed tips or defected media regions. It is because usually the device can notice which tips or regions are failed so such failures result in data "erasures" instead of data "errors". The extra space cost for burst bit erasures correction is about 3.13% of effective storage.

Because tip crashes and media defects happen rarely, we can even use another outer encoder across multiple basic data reliability units to protect against such burst bit erasures and achieve better space efficiency. This scheme is similar to the Cross-Interleaved RS Code (CIRC) used in CD-ROM [17] and the Reed-Solomon product code used in DVD [26].

There are two disadvantages of using long ECC codewords. First of all, using longer codewords and more parity bits increases the complexity and the cost of encoding and decoding circuits. The operations during encoding and decoding are more expensive and the bandwidth decreases. Secondly, using long ECC codewords increases the minimum size of accessible data units of the device.

5 Reliable Storage Building Blocks — MEMS Storage Enclosures

Thanks to its non-volatility, MEMS-based storage can replace or complement disks in storage systems. In general, disks have much higher storage capacities than MEMS storage devices. The expected capacity of a single MEMS device is 3–10 gigabyte [28]. In contrast, the capacities of state-of-the-art hard disks range from 18–300 GB for server SCSI disks, 40–250 GB for desktop IDE disks, to 20–60 GB for laptop IDE disks [31]. Thus, storage systems require one to two orders of magnitude more MEMS devices than hard disks to meet their capacity requirements. Correspondingly, more connection components, *i.e.* buses and interfaces, are also needed. These can significantly undermine system reliability and increase system costs.

The advance of the magnetic disk technology is not well-balanced. The increase in disk capacity noticeably outpaces the increase in bandwidth [16]. Thus disk rebuild times are becoming longer, during which a a subsequent disk failure (or a series of subsequent disk failures) can result in data loss. Because MEMS devices are expected to have at least as high, if not higher, bandwidths as hard disks and their capacities are limited, device rebuild times are significantly shorter for MEMS devices than for disks, which can in turn reduce the vulnerability window length thus improve system reliability. Also the entry cost of MEMS devices is expected to be several times lower than hard disks so it is flexible to add on-line spare MEMS devices in storage systems to improve their reliability.

Because of the reliability and cost concerns, we believe that multiple MEMS devices should be integrated into one MEMS storage enclosure under one interface and organized as RAID-5. We choose RAID-5 as the data redundancy scheme because of its reliability, space efficiency, and wide acceptance.

The role of MEMS storage enclosures in storage systems is exactly the same as disks' — providing reliable persistent storage. A controller manages MEMS devices in an enclosure and exposes a linear storage space through the interface. MEMS enclosures are the basic building block of MEMS-based storage systems, just as disks in disk-based systems.

Besides data and parity devices, a MEMS enclosure also have several on-line spare MEMS devices to improve its overall reliability, durability, and economy. The controller is able to detect device failures in seconds or minutes. As long as there are spare devices, data recovery can start immediately without replacement ordering and human interferences, which significantly reduces the window of data vulnerability and the chances of human errors thus improves the MEMS enclosure reliability.

When an enclosure runs out of spares, it can notify the host system, the maintenance personnel and/or the end users by sending signals and triggering visible lights. The enclosure can be either upgraded by a new enclosure or replenished with new spares to increase its life time, depending upon users' decisions. Although an enclosure without spares can still tolerate one more failure thanks to the RAID-5 organization, a preventive replacement/repair strategy is still desirable because it can significantly improve the system reliability.

Adding on-line spares can reduce maintenance costs because maintenance for such enclosures can be less frequent. It can also improve the enclosure durability because an enclosure can tolerate several component failures in its economic lifetime.

6 Reliability of MEMS Storage Enclosures

MEMS storage enclosures are internally organized as RAID-5 with spares. Researchers traditionally approximate the lifetimes of RAID-5 systems as exponential and use Mean Time To Failure (*MTTF*) to describe their reliability [7, 13, 25]. This approximation is accurate enough because the lifetimes of the system components are also modeled as exponential and failed components can be replaced in time, *i.e.* the system is repairable. Thus, with failed device replacement, MEMS enclosures share similar reliability characteristics with RAID-5 systems and their lifetimes can also be modeled as exponential.

However, without failed device replacement, the life-

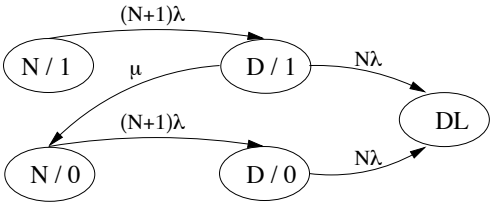


Figure 3: Markov model for a MEMS storage enclosure with N data and one parity devices and one dedicated spare. The enclosure can be in three modes: normal (N), degraded (D), and data loss (DL). We assume that MEMS lifetimes are independent and exponential with mean $MTTF_{mems} = 1/\lambda$. Recovery times of failed devices are also independent and exponential with mean $MTTR_{mems} = 1/\mu$. The numbers in the figure indicate how many spare MEMS devices the enclosure still has.

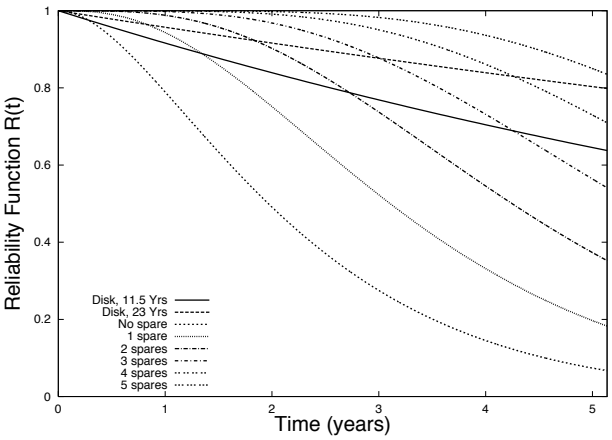


Figure 4: Reliability of MEMS storage enclosures using dedicated sparing without failed device replacement in five years.

times of MEMS enclosures can be viewed as two stages: the reliable stage with spares and the unreliable stage without spares. When it still has spare devices, a MEMS enclosure can be as reliable as RAID-5 systems with very short rebuild times; when spares run out, the enclosure becomes unreliable because any two successive device failures can result in data loss. Thus the lifetimes of MEMS enclosures without replacement cannot be simply modeled as exponential.

6.1 Reliability without Replacement

We first study the reliability of MEMS storage enclosures with dedicated spares in an idealistic but simple situation. The spare devices do not participate in request services during normal operations. We only consider the reliability of MEMS devices; other components in the enclosures are perfect. Failed MEMS devices are not replaced.

We assume that a MEMS enclosure contains 19 data, one parity, and k dedicated spare devices. The usable capacity of the enclosure is 60 GB because the capacity of a single MEMS device is 3.2 GB (see Table 1). We assume the data rebuild times from a failed device to an on-line spare are exponential with mean $MTTR_{mems} = 0.25$ hour. It is a very conservative estimation, considering the high bandwidth (76 MB/s) and relatively small capacity (3.2 GB) of MEMS: we only use less than 5% of the device bandwidth for data recovery.

Unfortunately there is no data on the reliability of MEMS-based storage devices because they are still being developed and not commercially available yet. Only limited literatures on the reliability of microelectromechanical systems are publicly available today. Douglass [9, 10] reported and estimated the Mean Time Between Failure (*MTBF*) of commercialized Digital Micromirror Devices (DMD), an MEMS-based digital imaging technology developed by Texas Instruments (TI), was 650,000 hours (74 years).

For simplicity, we assume that MEMS-based storage devices have exponential lifetimes. Researchers and engineers of MEMS-based storage expect that MEMS storage devices are more reliable than disks [5, 15]. Thus we further assume the Mean Time To Failure (*MTTF*) of MEMS devices is 200,000 hours (23 years). This number is partially supported by Douglass [9, 10]. For the purpose of comparison, we assume the lifetimes of commodity disks and "better" disks are also exponential with means of 100,000 and 200,000 hours, respectively. While the exact reliability numbers depend upon these assumptions, the techniques themselves do not.

Figure 3 gives the Markov model for a MEMS storage enclosure with N data and one parity devices and one dedicated spare. The Mean Time To Failure (*MTTF*) of systems with s dedicated spares is

$$MTTF \doteq \frac{s+1}{(N+1)\lambda} + \frac{1}{N\lambda},$$

where $1/\lambda$ is the mean of lifetimes of MEMS devices. Thus, $MTTF$ of MEMS enclosures with zero to five spares are 2.3, 3.5, 4.6, 5.8, 6.9, and 8.1 years respectively, which are surprisingly low.

Although with low *MTTF*, MEMS enclosures with several spares can be more reliable than single devices with *MTTF* as high as 200,000 hours (23 years), even though the enclosures are not repaired in their economic lifetimes (3–5 years), as shown in Figure 4. Figure 4 illustrates the reliability functions of MEMS enclosures without repairs and disks. We assumes the lifetimes of disks with different quality are 11.5 and 23 years, respectively. The

reliability function $R(t)$ of an individual system is defined as the probability that the system survives for any life time t given that it is initially operational [32]:

$$R(t) = Prob(lifetime > t \mid initially\ operational).$$

Because we use Markov models for the MEMS enclosure reliability behaviors, $R(t)$ can be expressed as:

$$\begin{aligned} R(t) &= \sum_{state\ i,\ i \neq DL} P_i(t) \\ &= 1 - P_{DL}(t) \end{aligned}$$

where $P_i(t)$ is the probability that the system is in state i after time t and DL is the state of data loss.

Figure 4 indicates that with 3–5 dedicated on-line spares a MEMS enclosure is more reliable than a single device with $MTTF = 23$ years in the first 3–5 years, even though fail devices in the enclosure are not replaced. For instance, the probability of data loss due to the failure of a MEMS enclosure with five spares in the first three years is 1.75%, much better than 12.31% of a single disk with *MTTF* of 23 years. However, when it runs out of spares, the enclosure becomes unreliable and the probabilities of data loss due to enclosure failure in one month and one year are 0.235% and 21.06%, respectively. Therefore, the lifetimes of MEMS enclosures without replacement cannot be simply modeled as exponential and we should focus on their reliability functions, instead of *MTTF*, in their economic lifetimes, say 3–5 years.

Fortunately, the host system, maintenance personnel, and/or end users can notice when an enclosure enters its unreliable stage then schedule a repair or replacement in time. Note that MEMS enclosures are only the building block of storage systems. All the data on a "sick" enclosure can be replicated to an on-line spare enclosure within one hour, assuming 17 MB/s bandwidth consumption, which is only $\frac{17}{76 \times 19} = 1.2\%$ of the aggregated bandwidth of the MEMS enclosure.

6.2 Reliability with Replacement

When they run out of spares, MEMS enclosures can notify the host systems, the maintenance personnel, and/or the end users so that a maintenance visit can be scheduled. There are two strategies for replacing failed devices in MEMS enclosures: the preventive strategy schedules replacement right after spares run out and the mandatory strategy schedules replacement only when the enclosures operate in the degraded RAID-5 mode without any spares. Figure 5 shows the Markov model for a MEMS enclosure with N data and one parity devices and one dedicated spare with replacement, in which ρ_0 and ρ_1 are the mandatory and preventive replacement rates, respectively. We assume that the failed component replacements are independent and exponential.

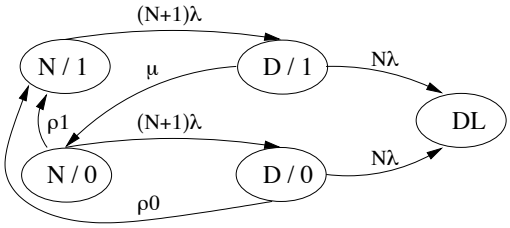


Figure 5: Markov model for a MEMS storage enclosure with N data and one parity devices and one dedicated spare. Failed component replacements are independent and exponential. The mandatory and preventive replacement rates are ρ_0 and ρ_1, respectively.

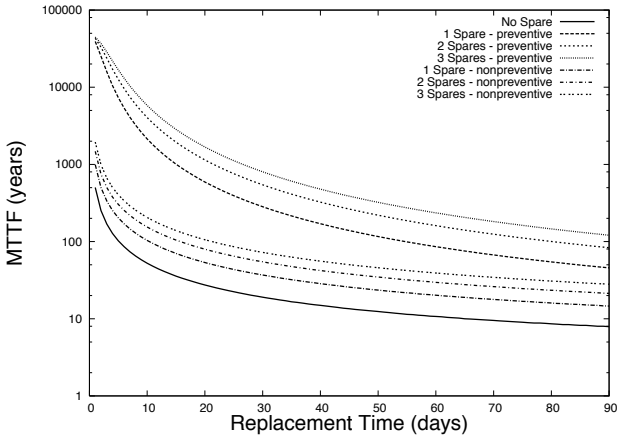


Figure 6: *MTTF* of MEMS storage enclosures using dedicated sparing under different replacement strategies and replacement rates, as represented as ρ_0 and ρ_1 in Figure 5. We set $\rho_0 = \rho_1 > 0$ in the preventive replacement strategy and $\rho_0 > 0$ and $\rho_1 = 0$ in the mandatory/nonpreventive replacement strategy.

Preventive replacement can significantly improve the reliability of MEMS enclosures because they can still tolerate one more failure during the replacement time, typically in days or weeks, thanks to their internal RAID-5 organization. Mandatory or nonpreventive replacement postpones enclosure repairs as late as possible so it can reduce the maintenance frequency during the enclosures' lifetimes. However, nonpreventive replacement makes users exposed to higher risks of data loss or unavailability.

Figure 6 shows *MTTF* of MEMS storage enclosures with different numbers of dedicated spares under different replacement strategies and replacement rates, ranging from one day to three months. We fix the number of data devices ($N = 19$) and the average data recovery

time to on-line spares ($1/\mu = 15$ minutes). Clearly, on-line spares with preventive replacement can dramatically increase *MTTF* of MEMS enclosures, about one to two orders of magnitudes higher than enclosures without on-line spares, under the same replacement rate because preventive replacement allows MEMS enclosures tolerate one more failure during the replacement time. Without preventive replacement, the reliability improvement by on-line spares is less impressive.

The reliability (*MTTF*) of MEMS enclosures is heavily dependent on how fast failed devices can be replaced: when the average replacement time increases from one day to one month, *MTTF* of enclosures can drop by one to two orders of magnitudes. Compared to nonpreventive replacement, preventive replacement can reduce replacement urgency under the same reliability requirement, as shown in Figure 6.

The number of active data devices N and the average data recovery rate to on-line spares μ also have impacts on MEMS enclosure reliability. Figures 7(a) and 7(b) show *MTTF* of MEMS enclosures, with one dedicated spare and using preventive replacement, as a function of N and μ given that the average failed device replacement times are one day and one week, respectively. We vary N from 19 to 23 and μ from 4 (15 minutes) to 30 (2 minutes), which are reasonable ranges for MEMS enclosures under consideration.

In general, *MTTF* decreases with the increase of the number of active data devices N, and with the decrease of the average data recovery rate μ. Note that the changes of *MTTF* under the specified ranges of N and μ are within four to five times. Thus, N and μ have less profound impacts on *MTTF* than the average device replacement rates, ρ_0 and ρ_1, as shown in Figure 6.

Given a relatively large failed device replacement time (one week on average), *MTTF* mostly relies on N instead of μ, as shown in Figure 7(b). When replacement tends to be postponed, the risk of data loss during the short data reconstruction time is neglectable, compared to the risk of data loss during the long replacement period. However, the length of the data reconstruction time becomes more relevant when the replacement time becomes shorter (one day on average), as shown in Figure 7(a). The enclosure reliability always decreases with the increase of the number of active data devices N.

6.3 Reliability of Distributed Sparing

In fact, spare storage in MEMS enclosures can also be organized in a distributed fashion. In distributed sparing [22], client data, parity data, and spare space are evenly distributed on all of the devices in the enclosure. This technique can provide better performance than dedicated sparing in the normal and data reconstruction modes [21, 22, 33]. Compared to dedicated sparing, distributed sparing can reduce data reconstruction times thus reduce the window of vulnerability and the risk of data loss because distributed sparing needs to reconstruct less data than dedicated sparing and its data reconstruction can be processed in parallel from and to all devices, avoiding the serialized reconstruction problem in dedicated sparing. However, distributed sparing utilizes more devices, which may undermine the overall enclosure reliability. Figure 8 gives a well-known layout of distributed sparing.

MEMS 1 MEMS 2 MEMS 3 MEMS 4 MEMS 5 MEMS 6

D1	D2	D3	D4	P1	S1
D6	D7	D8	P2	S2	D5
D11	D12	P3	S3	D9	D10
D16	P4	S4	D13	D14	D15
P5	S5	D17	D18	D19	D20
S6	D21	D22	D23	D24	P6

(a) Before any failure

MEMS 1 MEMS 2 MEMS 3 MEMS 5 MEMS 6

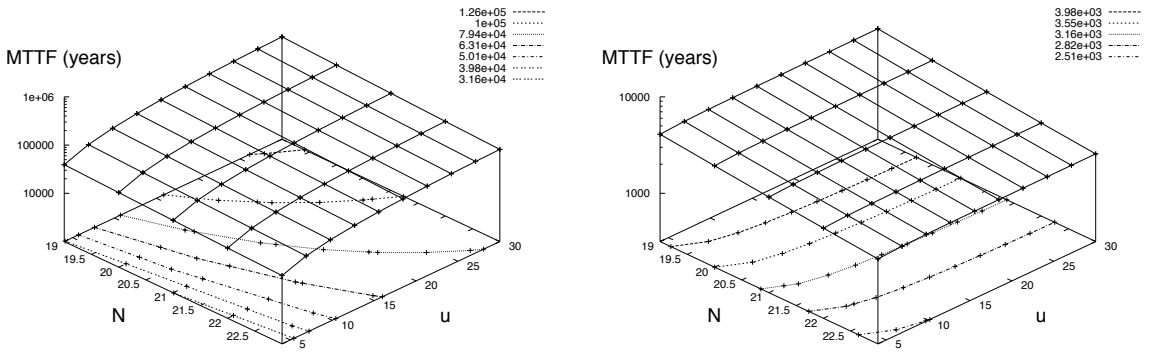
D1	D2	D3	P1	D4
D6	D7	D8	P2	D5
D11	D12	P3	D9	D10
D16	P4	D13	D14	D15
P5	D18	D17	D19	D20
D23	D21	D22	D24	P6

(b) After MEMS 4 fails

Figure 8: Distributed sparing.

Figure 9 shows the Markov model for a MEMS enclosure with N devices using distributed sparing. Because a MEMS enclosure generally stays in the data reconstruction modes for very short periods of time, we can safely merge the reconstruction modes to the normal modes by adding transitions directly from the normal modes to the data loss mode with small probabilities, $N\lambda q_j$, to simplifying our calculations. The probability of data loss during the data reconstruction time t_r when $j-1$ devices still survive, q_j, is $1-e^{-(j-1)\lambda t_r} \doteq (j-1)\lambda t_r$, where t_r is always no longer than its counterpart in dedicated sparing.

Distributed sparing and dedicated sparing can provide comparable or almost identical reliability to the MEMS



(a) One-day average replacement rate

(b) One-week average replacement rate

Figure 7: Contour figures for *MTTF* of MEMS storage enclosures with N data devices and one dedicated spare under failed device replacement cycles of (a) one day and (b) one week. Preventive replacement is assumed. Different data recovery rates to the on-line spare, μ, are also examined.

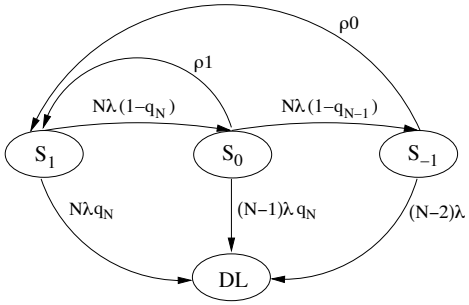


Figure 9: Markov model for a MEMS storage enclosure with N devices using distributed sparing. We assume that MEMS lifetimes are independent and exponential with mean $MTTF_{mems} = 1/\lambda$. We distinguish states $S_k, S_{k-1}, \cdots, S_0, S_{-1}$ where the index indicates the number of virtual spare devices left. State S_{-1} is the state in which the parity data is already lost. *DL* is the state of data loss. The probability of data loss during data reconstruction when $j-1$ devices still survive is q_j. Failed component replacements are independent and exponential. The mandatory and preventive replacement rates are ρ_0 and ρ_1, respectively.

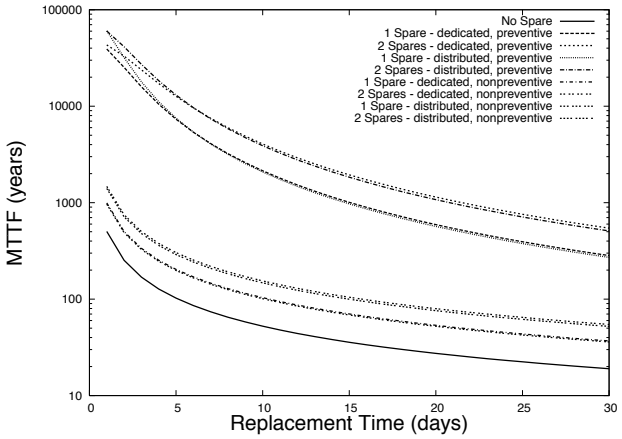


Figure 10: *MTTF* of MEMS storage enclosures using either dedicated or distributed sparing under different failed device replacement rates, as represented as ρ_0 and ρ_1 in Figures 5 and 9. We set $\rho_0 = \rho_1 > 0$ in the preventive replacement strategy and $\rho_0 > 0$ and $\rho_1 = 0$ in the mandatory/nonpreventive replacement strategy.

enclosure configurations under examination. Figure 10 compares *MTTF* of MEMS storage enclosures using either dedicated or distributed sparing with different numbers of spares under different failed component replacement rates, ranging from one day to three months. The storage capacity available to users is equivalent to the total storage of 19 MEMS devices, 60 GB. We set the data recovery rates to on-line spares in distributed sparing higher than those in dedicated sparing.

Distributed sparing requires less time to reconstruct data to on-line spares, which can improve reliability; on the other hand, it involves more active devices, which can undermine reliability. These two effects can balance each other, as shown in Figure 10 and Figures 7(a) and 7(b). In particular, dedicated sparing and distributed sparing provide almost identical MTTF to MEMS enclosures under the mandatory/nonpreventive replacement strategy. Typically, the average device replacement time is in days or weeks and the average data reconstruction time to on-line spares is in minutes. Thus, without preventive replacement the risk of data loss during data reconstruction is neglectable compared to the risk during device replacement. Although distributed sparing has shorter data reconstruction times than dedicated sparing, it has no sig-

nificant impact on MEMS enclosure reliability. When preventive replacement is employed, the risk of data loss during data reconstruction is comparable to the risk under fast replacement because the replacement times are short and the enclosures can tolerate one more failure during the replacement periods. Thus, distributed sparing provides better reliability than dedicated sparing only under this situation, as shown in Figure 10.

6.4 Other Issues on MEMS Storage Enclosure Reliability

In Sections 6.1 and 6.2, we assume only MEMS devices in MEMS storage enclosures can fail and other components are perfect. Failures of MEMS devices are also assumed to be independent. In reality, data loss can be caused by various reasons, such as correlated device failures, shared component failures, system crash, unrecoverable bit errors, and so on. We simply follow the failure analysis in [7, 29] and present the results here.

Like disks, MEMS device failures tend to be correlated due to common environmental and manufacturing factors. Also alive devices in an enclosure after the initial failure generally have to service much heavier workloads than usual due to external requests and internal data reconstruction requests. For simplicity, we follow the assumption that each successive device failure is 10 times more likely than the previous failure until the failed device has been reconstructed [7]. Under this assumption, *MTTF* of MEMS enclosures with or without preventive replacement drops about 9–10 times. Figure 6 suggests that preventive replacement is more desirable than mandatory/nonpreventive replacement because it can still provide high reliability without emergent repairs.

We assume that all MEMS devices in an enclosure are attached to common power and data strings. Then the probabilities of data loss due to string and controller failures in the first three years are 0.52% and 2.59% respectively, assuming the power string has *MTTF* of 5×10^6 hours (571 years) and the RAID-5 controller has *MTTF* of 10^6 hours (114 years) [29]. It suggests that the controller more likely results in data loss than MEMS devices although it is much more reliable than a single MEMS device.

Reliability estimations, following the approaches in [7], illustrate that the probabilities of data loss due to uncorrectable bit errors and system crash followed by a MEMS device failure in the first three years are 0.14% and 0.18%.

7 Economic Issues

As persistent storage devices, MEMS storage enclosures are expected to operate reliably as long as possible, in particular in their economic lifetimes. Compared to their counterparts – disks, MEMS enclosures are repairable. We are interested in the impacts of on-line spares and replacement policies on their durability and demands for maintenance services.

Storage servers built on MEMS storage typically require thousands to tens of thousands MEMS devices to meet their capacity requirements. As in MEMS enclosures, on-line spares can improve reliability and potentially reduce maintenance costs in such large installations. Unlike MEMS enclosures, such systems require periodic maintenance services. Thus a better understanding on how on-line spares can affect routine maintenance schedules is beneficial.

7.1 Durability of MEMS Storage Enclosures

In MEMS storage enclosures, failed device replacement tends to be postponed as late as possible or even not replaced during their economic lifetimes to minimize maintenance costs and human interferences. This strategy raises questions on the durability of MEMS enclosures: how long they can work without repairs? How many times they need repairing in their economic lifetimes, say 3–5 years? How different replacement policies effect the maintenance frequency?

Again we consider a MEMS enclosure with 19 data and one parity devices and k dedicated spares. For simplicity, we assume that data reconstruction to on-line spares completes instantaneously as one active device fails. Let $p_n(t)$ be the probability that n MEMS devices in the enclosure have failed during the period of $(0,t]$. As discussed in [3],

$$p_n(t) = e^{-\lambda_N t}(\lambda_N t)^n \frac{1}{n!}, \tag{1}$$

where $\lambda_N = N\lambda$, N is the number of data and parity devices in the enclosure and $1/\lambda$ is *MTTF* of MEMS devices, which is 23 years. Thus, the probability that a MEMS enclosure confronts up to k failures during the period of $(0,t]$ is

$$\begin{aligned} P_k(t) &= \sum_{n=0}^{n=k} p_n(t) \\ &= \sum_{n=0}^{n=k} e^{-\lambda_N t}(\lambda_N t)^n \frac{1}{n!}. \end{aligned} \tag{2}$$

In other words, the enclosure can survive after time t with the probability of $P_k(t)$ as long as it can tolerate up to k

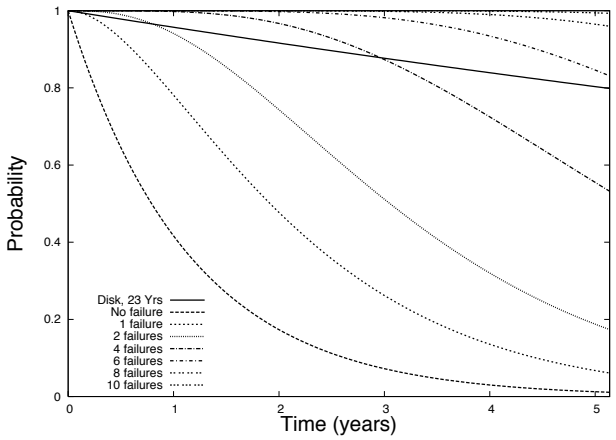


Figure 11: Probabilities that a MEMS storage enclosure has up to k failures during $(0,t]$.

failures. Figure 11 illustrates the probabilities that up to k failures occur in a MEMS storage during $(0,t]$.

Without any repairs, a MEMS enclosure with k spares can tolerate up to $k+1$ failures in its lifetime. With m repairs ($m \geq 1$), the enclosure can tolerate up to $k \times (m+1)$ failures under preventive replacement and $(k+1) \times (m+1)$ failures under mandatory/nonpreventive replacement before the $(m+1)th$ repair is scheduled. Here we assume enclosure repairs can be completed instantaneously because we are interested in how many times an enclosure has to be repaired during its economic lifetime, instead of its reliability.

For comparison, the probabilities that a disk with *MTTF* of 23 years can survive for more than one, three, and five years are 95.7%, 87.7%, and 80.3%, respectively. A MEMS enclosure with two spares has the chance of 98.8% to survive for one year without repair. The probability that an enclosure with five spares can survive for five years without repair is 84.6%. The chance that an enclosure with three spares under preventive replacement requires more than one repair during five years is 15.4%; instead, the chance for the same enclosure under nonpreventive replacement is only 3.5%. Adding one more spare can further reduce these probabilities to 3.5% and 0.6%, respectively. Obviously, preventive replacement trades more maintenance services for higher reliability, compared to mandatory replacement.

Figure 11 is almost identical to Figure 4 because the average data reconstruction time to on-line spares is very short in reality. Thus the assumption of immediate failure repairs in Figure 11 is quite accurate in calculating reliability without replacement. Therefore, we can quickly get approximations of the reliability functions of MEMS enclosures without replacement by using Equation 2, without solving messy ordinary differential equations.

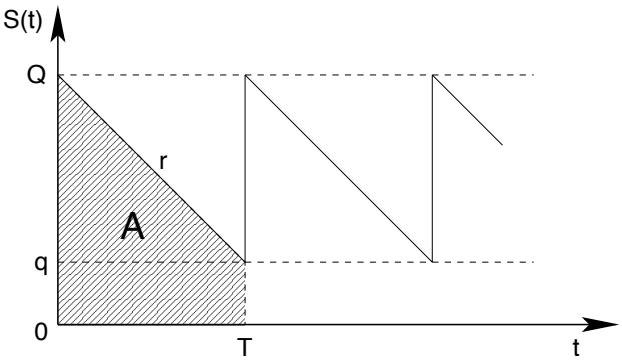


Figure 12: The number of spare disks $S(t)$ at time t.

7.2 Economy of On-line Spares in Storage Servers

Typically, a single state-of-the-art enterprise storage server can provide tens of terabyte (TB) physical storage. For instance, the physical capacity of the IBM TotalStorage Enterprise Storage Server (ESS) Model 800 can be up to 60 TB [19]. Tens of thousands of MEMS devices are required if such a server is built on MEMS storage. On-line spares can effectively improve the system reliability by shortening data reconstruction times. Meanwhile, on-line spares can potentially improve the system total cost of ownership by reducing the maintenance frequency thus the maintenance costs.

However, purchasing too many spare MEMS devices in advance can be a waste of money. The CMU MEMS design [5, 6] employs orthogonal magnetic recording techniques and uses a standard CMOS fabrication process to turn conventional interconnects of a CMOS integrated circuit into movable mechanical structures [12]. Thus, we can reasonably estimate that the cost per gigabyte of MEMS-based storage under mass production will continuously decline at a rate comparable to the disk's as the MEMS storage technology advances. Moreover, on-line spares increase electricity, cooling, and floor space costs. An better understanding of the balance between maintenance costs and the cost overheads of on-line spares is beneficial.

We assume that a storage system consists of N data devices and initially Q dedicated spare devices, shared by the whole system. Because the MEMS device capacity is quite limited, N is a large number typically. As in Section 6, the lifetimes of MEMS devices are exponential with the average of $1/\lambda$. When the number of spares drops below q, a repair is scheduled. Failed devices are replaced and on-line spares are replenished, back to Q.

Because the probability that a MEMS device fails in one day is $p = 1 - e^{-24\lambda}$ and N is a large number, the

average number of failures per day r is

$$r = N \times p = N \times (1 - e^{-24\lambda}). \quad (3)$$

So the maintenance interval T is

$$T = \frac{Q-q}{r}. \quad (4)$$

and the number of spares $S(t)$, decreases linearly with the rate r in an interval of T, as shown in Figure 12.

We further assume that the maintenance cost C_m, excluding device purchasing costs, is $c_1 + c_2 \times (Q-q)$, where c_1 is a constant overhead and c_2 is the repair cost per device. The depreciation cost per device per day is an constant c_3. Thus the total cost C_{total} in a maintenance interval T is the maintenance cost C_m plus the depreciation cost C_{dpr} in T:

$$\begin{aligned} C_{total} &= C_m + C_{dpr} \\ &= c_1 + c_2 \times (Q-q) + c_3 \times \int_0^T S(t)dt \\ &= c_1 + c_2 \times (Q-q) + c_3 \times (\frac{1}{2}(Q-q)T + qT). \end{aligned}$$

Because we are not interested in the total cost C_{total} in a variable interval T, we amortize C_{total} to a daily cost C_{daily}:

$$\begin{aligned} C_{daily}(T) &= \frac{C_{total}}{T} \\ &= \frac{c_1}{T} + \frac{c_2 \times (Q-q)}{T} + c_3 \times (\frac{Q+q}{2}). \quad (5) \end{aligned}$$

Combining Equations 3, 4, and 5 and letting $\frac{dC_{daily}}{dT} = 0$, we have

$$T = \sqrt{\frac{2c_1}{rc_3}} = \sqrt{\frac{2c_1}{Npc_3}}, \quad (6)$$

$$Q = \sqrt{\frac{2rc_1}{c_3}} + q = \sqrt{\frac{2Npc_1}{c_3}} + q. \quad (7)$$

It is interesting that the number of spares required in the system is proportional to the square roots of the number of active devices in the system and the ratio of the constant maintenance overhead to the daily device depreciation cost.

Consider a storage server with 32 TB physical capacity. It contains 10,000 active data MEMS devices, whose lifetimes are exponential with mean of 200,000 hours. Then the expected number of device failures per day is $r = 1.2$. Suppose the price of a MEMS device is $20 and its value drops to zero in five years. The the daily value drop is $0.011 per device. Because a on-line spare MEMS device consumes a tiny power drain in its inactive mode — about 0.05 Watts [5, 20, 28], the extra energy cost is neglectable, assuming that electricity costs $0.2/Kwh and cooling doubles the electricity cost. Thus the depreciation cost c_3 is $0.011 per device per day. The constant overhead cost of maintenance is $300 per visit. Using Equations 6 and 7, we can obtain $T = 178$ days and $Q - q = 213$ devices. Therefore, we can easily estimate the economically optimal maintenance schedule: the system should contain about 250 on-line spares (2.5% of the data devices), which makes a half-year maintenance schedule feasible. Clearly, the estimation technique is also suitable for disk-based storage systems as long as certain assumptions hold for those systems.

8 Concluding Remarks

Conclusion goes here.

Acknowledgments

We are grateful to ...

References

[1] AllAboutMEMS.com. MEMS applications. http://www.allaboutmems.com/memsapplications.html, 2004.

[2] ATA SMART feature set commands. Small Form Factors Committee SFF-8035. http://www.t13.org.

[3] U. N. Bhat and G. K. Miller. *Elements of Applied Stochastic Processes*. John Wiley & Sons, Inc., New Jersey, 3rd edition, 2002.

[4] M. Blaum, J. Brady, J. Bruck, and J. Menon. EVENODD: An efficient scheme for tolerating double disk failures in RAID architectures. *IEEE Transactions on Computers*, 44(2):192–202, 1995.

[5] L. Carley, J. Bain, G. Fedder, D. Greve, D. Guillou, M. Lu, T. Mukherjee, S. Santhanam, L. Abelmann, and S. Min. Single-chip computers with microelectromechanical systems-based magnetic memory. *Journal of Applied Physics*, 87(9):6680–6685, May 2000.

[6] L. R. Carley, G. R. Ganger, and D. F. Nagle. MEMS-based integrated-circuit mass-storage systems. *Communications of the ACM*, 43(11):72–80, Nov. 2000.

[7] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson. RAID: High-performance, reliable secondary storage. *ACM Computing Surveys*, 26(2):145–185, June 1994.

[8] R. D. Cideciyan, E. Eleftheriou, and T. Mittelholzer. Perpendicular and longitudinal recording: a signal-processing and coding perspective. *IEEE Transactions on Magnetics*, 38(4):1698–1704, July 2002.

[9] M. R. Douglass. Lifetime estimates and unique failure mechanisms of the digital micromirror device (DMD). In *Proceedings of the 36th IEEE International Reliability Physics Symposium*, pages 9–16, May 1998.

[10] M. R. Douglass. DMD reliability: a MEMS success story. In *Proceedings of SPIE*, volume 4980, pages 1–11, San Jose, CA, Jan. 2003. SPIE.

[11] R. H. Dunphy, Jr., R. Walsh, and J. H. Bowers. Disk drive memory. United States Patent 4,914,656, Apr. 1990.

[12] G. Fedder, S. Santhanam, M. L. Reed, S. Eagle, D. F. Guillou, M. Lu, and L. R. Carley. Laminated high-aspect-ratio microstructures in a conventional CMOS process. In *Proceedings of the IEEE Micro Electro Mechanical Systems Workshop*, pages 13–18, February 1996.

[13] G. A. Gibson. *Redundant Disk Arrays: Reliable, Parallel Secondary Storage*. PhD thesis, University of California at Berkeley, 1990.

[14] J. L. Griffin, S. W. Schlosser, G. R. Ganger, and D. F. Nagle. Modeling and performance of MEMS-based storage devices. In *Proceedings of the 2000 SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pages 56–65, Santa Clara, CA, June 2000. ACM Press.

[15] J. L. Griffin, S. W. Schlosser, G. R. Ganger, and D. F. Nagle. Operating system management of MEMS-based storage devices. In *Proceedings of the 4th Symposium on Operating Systems Design and Implementation (OSDI)*, pages 227–242, San Diego, CA, Oct. 2000. USENIX Association.

[16] J. L. Hennessy and D. A. Patterson. *Computer Architecture—A Quantitative Approach*. Morgan Kaufmann Publishers, 3rd edition, 2003.

[17] W. C. Huffman and V. Pless. *Fundamentals of Error-Correcting Codes*. The Cambridge University Press, 3rd edition, 2003.

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

[19] IBM Corporation. IBM TotalStorage Enterprise Storage Server (ESS) Model 800. http://www.storage.ibm.com/disk/ess/ess800/index.html, 2004.

[20] Y. Lin, S. A. Brandt, D. D. E. Long, and E. L. Miller. Power conservation strategies for MEMS-based storage devices. In *Proceedings of the 10th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS '02)*, Fort Worth, TX, Oct. 2002.

[21] J. Menon and D. Mattson. Comparison of sparing alternatives for disk arrays. In *Proceedings of the 19th International Symposium on Computer Architecture*, pages 318–329, Queensland, Australia, May 1992. ACM Press.

[22] J. Menon and D. Mattson. Distributed sparing in disk arrays. In *Proceedings of Compcon '92*, pages 410–416, Feb. 1992.

[23] R. R. Muntz and J. C. S. Lui. Performance analysis of disk arrays under failure. In *Proceedings of the 16th Conference on Very Large Databases (VLDB)*, pages 162–173, Brisbane, Queensland, Australia, 1990. Morgan Kaufmann.

[24] Nanochip Inc. Nanochip: Array nanoprobe mass storage IC. Nanochip web site, at http://www.nanochip.com/preshand.pdf, 1999.

[25] D. A. Patterson, G. Gibson, and R. H. Katz. A case for redundant arrays of inexpensive disks (RAID). In *Proceedings of the 1988 ACM SIGMOD International Conference on Management of Data*, pages 109–116. ACM, 1988.

[26] Pioneer Corporation. DVD technical guide. http://www.pioneer.co.jp/crdl/tech/index-e.html, July 1999.

[27] A. L. N. Reddy and P. Banerjee. Gracefully degradable disk arrays. In *Proceedings of the 21rd International Symposium on Fault-Tolerant Computing (FTCS '91)*, pages 401–408, Montreal, Canada, June 1991. IEEE Computer Society Press.

[28] S. W. Schlosser, J. L. Griffin, D. F. Nagle, and G. R. Ganger. Designing computer systems with MEMS-based storage. In *Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 1–12, Cambridge, MA, Nov. 2000. ACM Press.

[29] M. Schulze, G. Gibson, R. Katz, and D. Patterson. How reliable is a RAID? In *Proceedings of Compcon '89*, pages 118–123. IEEE, Mar. 1989.

[30] SCSI "Mode sense" code "Failure prediction threshold exceeded". American National Standards Institute. http://www.t10.org.

[31] Seagate Technology, Inc. Seagate Disc Product Datasheets. http://www.seagate.com/products/datasheet/, August 2004.

[32] D. P. Siewiorek and R. S. Swarz. *Reliable Computer Systems Design and Evaluation*. The Digital Press, 2nd edition, 1992.

[33] A. Thomasian and J. Menon. RAID5 performance with distributed sparing. *IEEE Transactions on Parallel and Distributed Systems*, 8(6):640–657, June 1997.

[34] J. W. Toigo. Avoiding a data crunch – A decade away: Atomic resolution storage. *Scientific American*, 282(5):58–74, May 2000.

[35] P. Vettiger, M. Despont, U. Drechsler, U. Urig, W. Aberle, M. Lutwyche, H. Rothuizen, R. Stutz, R. Widmer, and G. Binnig. The "Millipede"—More than one thousand tips for future AFM data storage. *IBM Journal of Research and Development*, 44(3):323–340, 2000.