A Dual Actuator Logging Disk Architecture

John A. Chandy
Department of Electrical and Computer Engineering
University of Connecticut
260 Glenbrook Road, U-2157
Storrs, CT 06269-2157 USA
email: john.chandy@uconn.edu

ABSTRACT
In this paper, we present a dual actuator logging disk architecture to minimize write access latencies. We reduce small synchronous write latency using the notion of logging writes, i.e. writing to free sectors near the current disk head location. However, we show through simulations that logging writes by itself is not sufficient to reduce write access latencies, particularly in environments with writes to new data and intermixed reads and writes. Therefore, we augment the logging write method with the addition of a second disk actuator. Our simulations show that the addition of the second actuator offers significant performance benefits over a normal disk over a wide range of disk access patterns, and comparisons to strictly logging disk architectures show advantages over a range of disk access patterns.

KEY WORDS
Disk architecture, Storage systems, Computer architecture

1 Background

Information is being created and consumed at an increasingly faster rate and studies have suggested that the amount of information stored digitally will continue to double every year for the foreseeable future[1]. The increasing need for information storage leads to a corresponding need for high-performance and reliable storage systems. However, the pace of development in storage technologies has not kept up with the rapid pace of development with other computer technologies. Specifically, computer processors have been improving in speed by a factor of two every 18 months, and while disk capacity has been growing exponentially as well, access times, the most critical measure of disk speed, have only been improving 10% per year. This discrepancy leads to storage systems increasingly becoming the bottleneck in computer systems particularly in applications such as transaction processing and database systems. In this paper, we introduce an architecture to address some of these lags in disk access speed.

Before going into more detail, it is instructive to briefly describe current disk architectures. A hard disk drive is composed of one or many circular platters or disks coated with a magnetic material that stores the 1's and 0's of the data. Positioned above and/or below each platter is a read/write head responsible for altering the orientation of the magnetic material in the case of a write, and sensing the orientation in the case of a read. In order to access all portions of the platter, the head can be moved radially across the platter by an actuator and secondly the platter itself is spun around a spindle at a fixed rotational speed to allow access to all angular positions. Thus, the two main components of the time to access a disk are the seek time, the time for the actuator to move or seek to the appropriate radial position, and the rotational latency, the time for the desired angular position to spin around under the head. The seek time is determined by the radial distance of the desired data from the current head position and the speed of the actuator. Average seek times are on the order of 3-4 milliseconds for current high-performance disks. The rotational latency is largely determined by the rotational speed of the disk. Current disks have rotational speeds of up to 15000 rpm meaning an average rotational latency of 2 milliseconds.

2 Dual Actuator Disk with Logging

The architecture that we are proposing reduces the average seek times for writes to near zero and thereby reduces overall disk access times for mixed disk access patterns. There are two parts to the architecture. The first is the reduction of synchronous write seek time using the notion of write logging or eager writes. The second part is the addition of a second actuator and set of heads to service reads.

Logging optimizes writes by writing data to empty locations near the current disk head location. The assumption is that if the disk is doing only writes, the disk head needs to seek only slightly, if at all, to write the new data, thus eliminating the seek time. The idea of write logging is not new and previous work has shown the effectiveness of the method in certain scenarios [2, 3, 4, 5, 6, 7]. However, it is only an effective strategy when the disk access pattern is mostly update writes. In other words, if the access pattern consists of random reads mixed in with writes to new data, logging writes no longer performs as well. With mostly update writes, i.e. writes to old date, the number of available free sectors in the active area of the disk remains the same. Thus, the disk allocator will always be able to find free sectors near the current location of the disk head. However, if there are writes to new data, the number of free sectors will reduce. In addition, the random reads in the disk access pattern force the disk head into areas of the disk where there are fewer free sectors as compared to the edge of the active area.

In order to address this problem, we propose the addition of a second actuator and set of disk heads to the hard drive. This second actuator will be dedicated to reads, thus allowing the logging write head to remain in regions where there are more available free sectors. The use of multiple disk actuators has been suggested in early literature [8], but there have been few commercial implementations, namely the IBM 3340 and Conner Chinook disk drives [9, 10]. The natural configuration for placing two actuators in a disk drive enclosure is to place them diagonally opposite each other as shown in Figure 1. We call this two-actuator disk architecture a Dual Actuator Disk (DAD). The addition of a second actuator as suggested in previous work allows reads and writes to be accelerated. In fact, the overall access time is roughly halved because of the ability to read data at two different radial and angular positions. However, a second actuator alone can not completely eliminate seek times. With the addition of write logging, however, the second disk head can now be used to service reads while the first head can be dedicated to doing writes. This architecture now guarantees near-zero-access writes regardless of the read behavior while at the same time providing access times for reads equivalent to a normal disk. We call this architecture a Dual-Actuator Logging Disk (DALD).

It might be argued that practical DADs are difficult to build and thus the lack of commercial implementations in spite of the obvious performance benefits. One of the main challenges is the difficulties in simultaneously tracking two separate mechanical arms in a single enclosure. In light of that, in our architecture we assume that only one arm is seeking and accessing data at a time. We, therefore, do not assume simultaneous tracking of the two arms in our simulations. Another problem is the ability to efficiently write data with one head and read with another. Though difficult, this problem is manageable as seen by the Conner Chinook implementation [10].

Cost has also been a significant reason as well. We believe that the DALD architecture that we have presented would actually have cost advantages over a normal DAD. In a DAD, each arm is identical to that found in a normal disk - i.e. will have both read and write heads integrated onto the same arm. Manufacturing these integrated read/write heads is difficult and complicates the design. In our DALD architecture, one arm will only have a read head on it, and the second arm will only have a write head. This simplifies the design of the head assembly since the arm does not need the complexities of an integrated read/write head. Potentially, the simpler heads would also lead to lighter heads and arms and thus faster seek times.

It is also useful to compare this architecture with disk arrays, since one of the reasons that DAD drives have not been commercially successful is their high cost relative to an array of cheap drives [11]. The argument has been that multiple cheap disks in a disk array can deliver better performance and better cost than a single high performance disk. While disk arrays will deliver better throughput than a single drive, it can not offer the same response time as a single drive. In general, the total response time will be related to the slowest of all the drives in the array. This has been demonstrated in previous performance studies comparing disk striping to single disks [12, 13, 14].

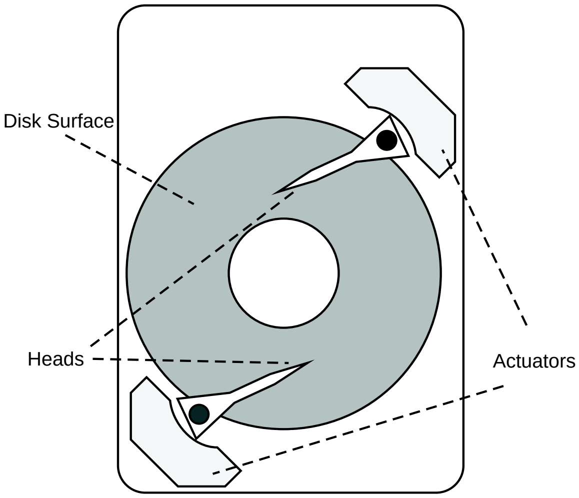

Figure 1. Disk Assembly with Two Arms

For certain applications, such as interactive systems, real-time systems, and certain types of transaction processing, response time is just as important if not more important than throughput. Moreover, in such applications, writes can be a significant portion of the access pattern. For that reason, we have focused the design of the DALD drive to address these sets of applications. For such applications, disk arrays can not deliver equivalent response time performance.

In the following section, we present simulation results that demonstrate the performance benefits of a DALD drive.

3 Simulations

To validate the proposed architecture, we performed simulations of five architectures - a disk with a single actuator, a disk with two actuators (DAD), a disk with single actuator and logging, a two-disk striped disk array and finally, our proposed architecture, the dual actuator disk with logging (DALD).

Simulations were performed with a process simulator using the IBM 18ES as the base drive. Parameters for the drive are shown in Table 1. It was assumed to have a single zone and a seek model that is linear with a startup latency. The linear seek model is not ideal, but it has been shown to have only a mean deviation from actual behavior of only about 9% [15].

Before taking measurements, the simulated disk was

Variable	Value
Single cylinder seek time	1.086 ms
Full strobe seek time	12.742 ms
Head switch time	0.062 ms
Rotation speed	7200 rpm
Rotation time	8.333 ms
Write settling delay	0.126 ms
Number of cylinders	11474
Number of surfaces	5
Number of sectors per track	312

Table 1. IBM 18ES Disk Parameters

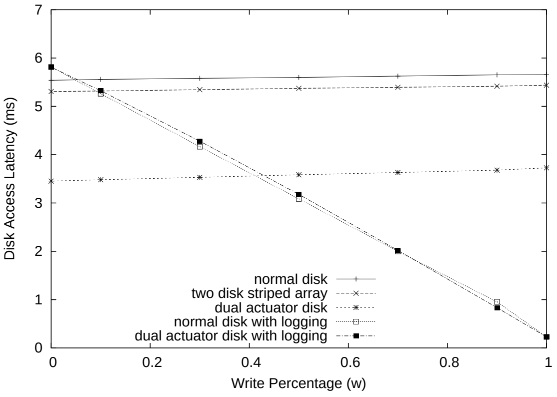

Figure 2. IBM 18ES Disk Access Latency ($\eta = 0$)

initialized with writes to the first 1000000 sectors. The results shown are for the post-initialization phase where we issue 500000 read and write requests to the disk distributed with a Poisson process interarrival rate of 10 requests per second. Each request is a 4K block (8 sector) read or write randomly and equally distributed across the active region of the disk. We use a synthetic load to drive the simulation rather than actual traces in order to allow us to study the effect of varying the write frequency as well as the effect of writes to new data.

Figures 2, 3, and 4 show simulations for the five disk architectures varying the write frequency with different values of the new write frequency, η. The new write frequency is the percentage of writes that are writes to new data. The graphs show the clear advantage of two-arm disks and both type of logging disks over both a normal disk and a two disk striped array. The reason the disk array response times are comparable to a single disk is that we have assumed the two disks in the array are synchronized.

For read-intensive access patterns, it is clear that a two-arm disk is preferable to either type of logging disk. However, as the write frequency increases, a logging disk becomes more preferable. The breakeven point is lower for the two-arm logging disk. With the presence of buffer caches in most operating systems, it is not unreasonable to expect that the disk access pattern will be skewed towards writes. Results presented by Roselli, et al. show that in file server usage with a reasonably large cache, the write fraction varied from 42% to 64% [16]. Roselli did not measure certain write intensive applications such as logfile updates, transaction processing, and data collection which may be expected to have higher cache hit rates for reads and thus more of a bias towards writes. Therefore, in these workloads with high write frequencies, the use of logging disks and in particular, two-arm logging disks is desirable.

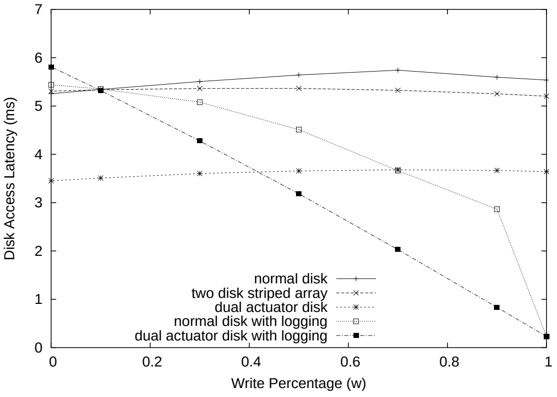

Figure 3. IBM 18ES Disk Access Latency ($\eta = .5$)

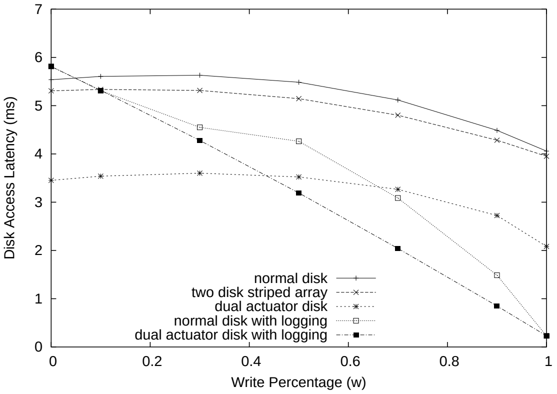

Figure 4. IBM 18ES Disk Access Latency ($\eta = 1$)

For all nonzero values of η it can also be seen that the logging drives with two actuators performs better. This is as expected, since as the frequency of writes to new data goes up, the single actuator logging drive has a harder time finding free sectors near the current head position. Further simulations of the effect of the new write frequency on the two logging disk architectures can be seen in Figures 5, 6, 7, 8, and 9, where we did simulations varying values of η for different values of the write frequency, w. When, $w = 0$ or $w = 1$, there is no substantial difference between a single actuator or dual actuator logging drive. This is because when $w = 0$, the drive is doing all reads and thus the second actuator in the DALD drive contributes nothing. When $w = 1$, the drive is doing all writes; so there are no reads to move the single actuator disk head away from the

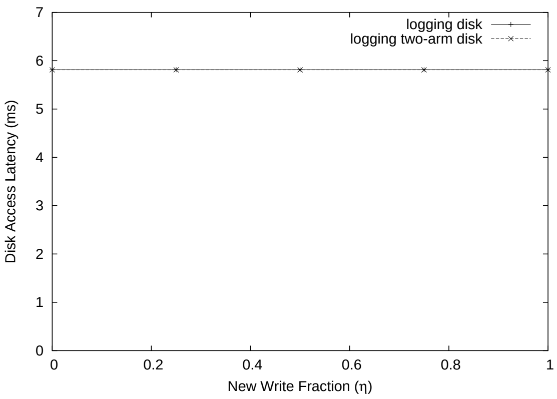

Figure 5. Disk Access Latency varying η (w=1)

available free sectors, thus negating the advantage of the DALD drive. However, for values of w in between 0 and 1 and all nonzero values of η, we see that the second arm reduces the access time substantially. When $\eta = 0$, these graphs show that a single arm logging disk has equivalent performance to a two-arm logging disk. The reason for this anomaly is that during the initialization phase of the simulation, the active area of the disk is not completely filled. When $\eta = 0$, these free sectors never get used, so the steady state assumption of a full active area is never reached.

In practice, $\eta = 0$ would not be expected to occur since all workloads will have a certain degree of writes to new data. The examples given above, logfile updates, transaction processing, and data collection will all have a very high degree of writes to new data. In addition, modern filesystems have a feature called snapshots whereby existing blocks are not deleted on an update write. Instead, the old blocks are retained and copied on update. The feature allows easy rollback to previous checkpoints of a filesystem. In such systems with a snapshot feature enabled, it would then be expected that a very high percentage of writes to disk will be writes to new data blocks. These systems with even very minimal values of η, the difference between a two-arm logging disk and single arm logging disk can be clearly seen.

It is instructive to determine the effect of disk drive parameters on the results, and so we have performed the same simulations using a Seagate Cheetah 9LP drive as the base drive. Figures 10, 11, and 12 show results for different values of η. The main consclusions are the same, i.e. better performance of DALD drives for high write frequencies and better performance of dual actuator drives at high read frequencies. However, the scale is different - particularly the difference in performance of the two logging disk architectures. For the Cheetah 9LP the difference is not as substantial as what was seen with the IBM 18ES drive.

At first glance, it may appear that this is due to the Cheetah's much higher rotational speed and thus lower rotational latency. However, in fact, the difference is actually due to the fact that the Cheetah has much fewer cylin-

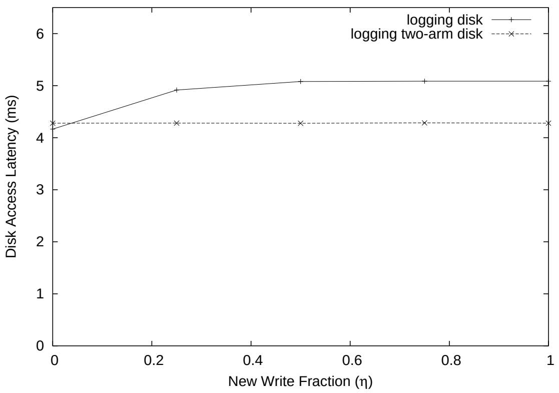

Figure 6. Disk Access Latency varying η (w=.7)

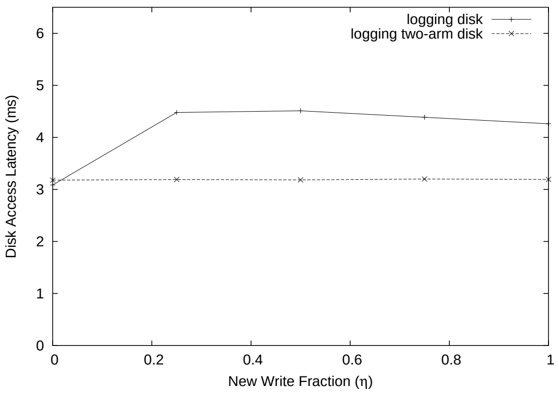

Figure 7. Disk Access Latency varying η (w=.5)

Variable	Value
Single cylinder seek time	0.831 ms
Full strobe seek time	10.627 ms
Head switch time	0.03 ms
Rotation speed	10045 rpm
Rotation time	5.973 ms
Write settling delay	0.461 ms
Number of cylinders	6962
Number of surfaces	12
Number of sectors per track	232

Table 2. Seagate Cheetah 9LP Disk Parameters

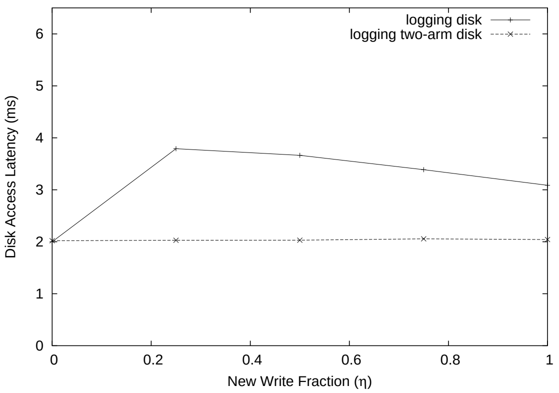

Figure 8. Disk Access Latency varying η (w=.3)

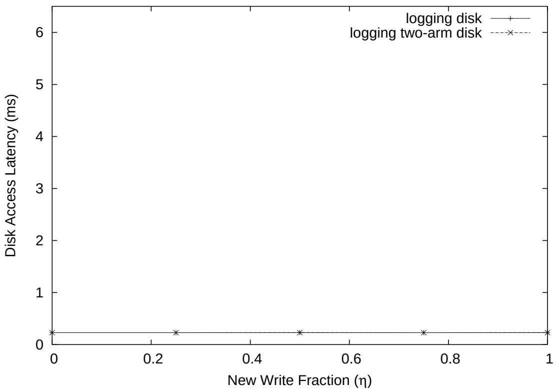

Figure 9. Disk Access Latency varying η (w=0)

ders and many more disk surfaces than the 18ES. It can be shown through analytical models that the primary determinant of the difference between a DALD drive and a logging disk is the number of cylinders. Without fully deriving these models, we can explain the reasoning as follows. As the number of cylinders increases, the probability increases that we will have to leave the current cylinder to find a free sector for a write in a logging disk. As this probability increases, the average access time of a logging disk increases as well since the seek time from cylinder to cylinder is significantly more than seeking within the same cylinder. A DALD drive's performance, however, does not depend on the number of cylinders since the write head is always on the last active cylinder.

In light of this behavior, we would expect that DALD drives will show a significant performance increase over a normal logging disk when the number of cylinders increases. As disk technology improves, this is likely to be more and more true as the track per inch densities increase leading to more and more cylinders per disk.

We have assumed that the system is under such high load that efficient compaction to free up sectors in a logging system is not feasible. Compaction and log cleaning is not as much of an issue with the two-arm logging system as with a single-arm system. A single-arm logging system depends on compaction in order to ensure that there are sufficient free blocks on each track. With the two-arm system that we have presented, writes are always done to the end of the active region where there are always free blocks. When the active region has reached the end of the disk, writes must occur within the active region where there are now free blocks. At this point, the two-arm system will suffer the same drawbacks as a single-arm system in terms of new writes occupying the available free blocks. But it will fare no worse than a single-arm system. Therefore, compaction and log cleaning need not be done continuously as with a normal logging system but only when the active region approaches the end of the disk.

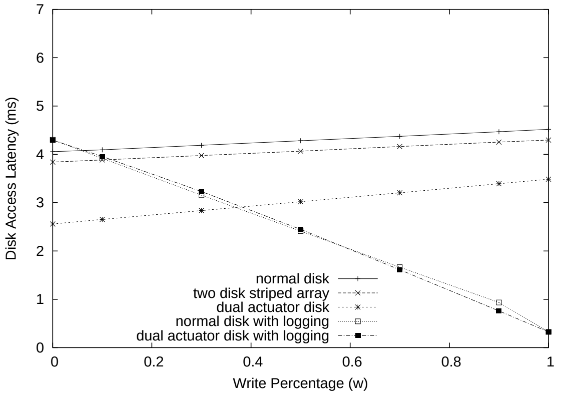

Figure 10. Seagate Cheetah 9LP Disk Access Latency ($\eta = 0$)

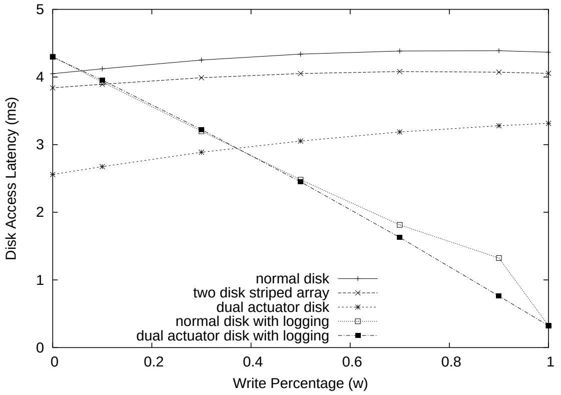

Figure 11. Seagate Cheetah 9LP Disk Access Latency ($\eta = .5$)

The simulations that we have presented show the advantages of a DALD drive for write intensive loads. With some intelligent controllers, the system could actually perform just as well as a non-logging DAD drive even for read-intensive workloads. If we outfit both arms of the DALD drive with read and write heads as with a normal drive, an intelligent controller could dynamically determine the workload characteristics and adaptively switch

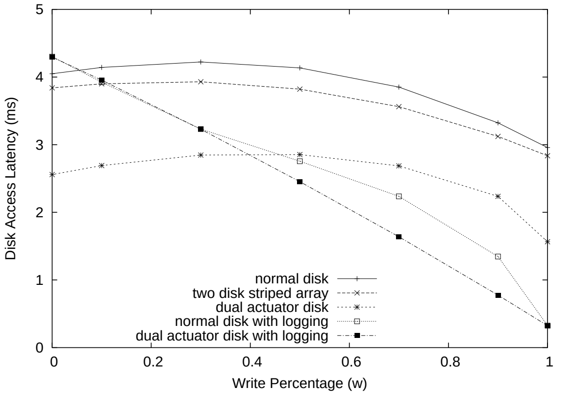

Figure 12. Seagate Cheetah 9LP Disk Access Latency ($\eta = 1$)

from DALD mode to DAD mode if it determines that the workload has become read intensive. This simple modification allows the DALD drive to deliver high performance response times for all workloads.

4 Conclusions

We believe that the proposed architecture will greatly improve disk access latencies for workloads that are write oriented. The results from this paper should encourage disk manufacturers to revisit the idea of dual actuator disks. Moreover, with logging and intelligent controllers, these disks should prove to be an ideal architecture for all disk access patterns. As mentioned in the introduction, the decrease of disk access times can have a significant effect on computer system performance. We have demonstrated that logging disk architectures must take into account the effect of intermixed reads and writes. The simulations have also shown how a second disk actuator can help the performance of a logging disk.

References

[1] P. Lyman and H. R. Varian, "How much information?." http://www.sims.berkeley.edu/research/projects/how-much-info, 1999.

[2] R. M. English and A. A. Stepanov, "Loge: A self-organizing storage device," in *Proceedings of the USENIX Technical Conference*, Jan. 1992.

[3] C. Chao, R. English, D. Jacobson, A. Stepanov, and J. Wilkes, "Mime: A high performance parallel storage device with strong recovery guarantees," Technical Report HPL-CSP-92-9 rev 1, Hewlett-Packard, Palo Alto, CA, Mar. 1992.

[4] T.-C. Chiueh, "Trail: A track-based logging disk architecture for zero-overhead writes," in *Proceedings of International Conference on Computer Design*, Oct. 1993.

[5] L. Huang and T.-C. Chiueh, "Trail: Track-based logging in Stony Brook Linux," in *Proceedings of International Conference on Dependable Systems and Networks*, June 2002.

[6] R. Y. Wang, T. E. Anderson, and D. A. Patterson, "Virtual log based file systems for a programmable disk," in *Proceedings of Symposium on Operating Systems Design and Implementation*, Feb. 1999.

[7] J. Menon, J. Roche, and J. Kasson, "Floating parity and data disk arrays," *Journal for Parallel and Distributed Computing*, vol. 17, pp. 129–139, Jan. 1993.

[8] A. J. Smith, "On the effectiveness of buffered and multiple arm disks," in *Proceedings of the International Symposium on Computer Architecture*, Apr. 1978.

[9] R. B. Mulvany, "Engineering design of a disk storage facility with data modules," *IBM Journal of Research and Development*, pp. 489–505, 1974.

[10] J. P. Squires, G. N. Bagnell, C. M. Sander, and K. M. Anderson, "Multiple actuator disk drive." Conner Peripherals, Mar. 1994. US Patent No. 5,293,282, WIPO Patent No. 9209077.

[11] D. A. Patterson, G. A. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks (RAID)," in *Proceedings of the ACM SIGMOD International Conference on Mangement of Data*, pp. 109–116, June 1988.

[12] D. Bitton and J. Gray, "Disk shadowing," in *Proceedings of International Conference on Very Large Data Bases (VLDB)*, pp. 331–338, Sept. 1988.

[13] P. M. Chen, G. A. Gibson, R. H. Katz, and D. A. Patterson, "An evaluation of redundant arrays of disks using an Amdahl 5890," in *Proceedings of the ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, pp. 74–85, May 1990.

[14] J. Gray, B. Host, and M. Walker, "Parity striping of disk arrays: Low-cost reliable storage with acceptable throughput," in *Proceedings of International Conference on Very Large Data Bases (VLDB)*, pp. 148–161, Aug. 1990.

[15] C. Ruemmler and J. Wilkes, "A trace-driven analysis of working set sizes," Technical Report HPL-OSR-93-23, Hewlett-Packard, Palo Alto, CA, Apr. 1993.

[16] D. Roselli, J. Lorch, and T. E. Anderson, "A comparison of file system workloads," in *Proceedings of the USENIX Technical Conference*, June 2000.