Data Placement Based on the Seek Time Analysis of a MEMS-based Storage Device

Zachary N. J. Peterson†
zachary@cse.ucsc.edu

Scott A. Brandt
scott@cse.ucsc.edu

Darrell D. E. Long
darrell@cse.ucsc.edu

Computer Systems Laboratory
University of California, Santa Cruz

Abstract

Reducing access times to secondary I/O devices has long been the focus of many systems researchers. With traditional disk drives, access time is the composition of transfer time, seek time and rotational latency, so many techniques as to minimize these factors, such as ordering I/O requests or intelligently placing data, have been developed. MEMS-based storage devices are seen by many as a replacement or an augmentation for modern disk drives, but algorithms for reducing access time for MEMS-based storage are still poorly understood. These devices, based on MicroElectroMechanical systems (MEMS), use thousands of active read/write heads working in parallel on a non-rotating magnetic substrate, eliminating rotational latency from the access time equation. This leaves seek time as the dominant variable. Therefore, new data layout techniques based on minimizing the unique seek time characteristics of a MEMS-based storage device can be developed. This paper begins to examine the access qualities of a MEMS-based storage device, and based on experimental simulation, develops an understanding of the seek time characteristics of such a device. These characteristics then allow us to identify equivalent regions in which to place data for improved access.

1 Introduction

Modern disk drives can no longer keep up with the performance trends of integrated circuit (IC) technology. The RAM-to-disk memory hierarchy gap is increasing at a rate of 50% per year [4], creating a performance bottleneck in computer systems. To compensate for this behavior, many techniques based on limiting the seek and rotational latency of a disk drive have been developed to improve disk, and therefore, system performance [6, 13]. These techniques include placing data on disk, based on workload, to reduce the time to access that data [8, 9, 11].

Traditional rotating disks have a cylindracal layout that has just one degree of freedom, making data layout a one-dimensional problem. MEMS-based storage, however, uses a two-dimensional array of read and write heads that act in parallel on a non-rotating media sled. Applying analogous layout techniques to MEMS-based storage suggests that this approach will not take full advantage of the parallelism or additional degree of locality inherent in the device.

Griffin *et al.* have shown that for a limited class of MEMS devices, one-dimensional placement and scheduling can be applied efficiently [3]. In this work, data is placed on the MEMS-device in longitudinally sequential tracks, similar to tracks on a disk drive. However, this method is preferred only if a long longitudinal seek takes less time than a smaller latitudinal seek.

To better develop an understanding of the access time characteristics of a MEMS-device we have created a device simulator that generates seek times from any two points on the device, implementing a device motion model formulated at the University of California, Santa Cruz (UCSC) [7]. In brief, we model the physical and mechanical forces that affect

†Supported in part by the Institute for Scientific Computing Research at the Lawrence Livermore National Laboratory.

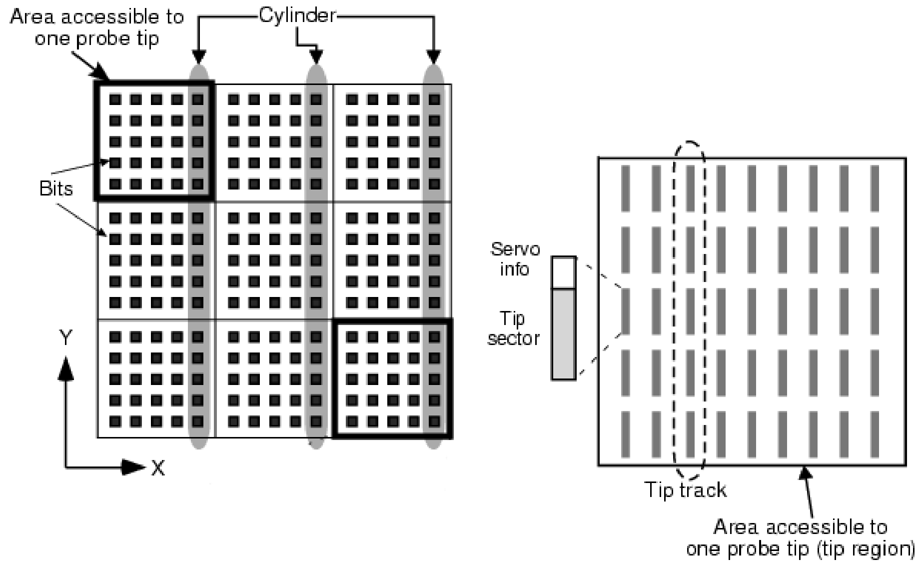

Figure 1: The low level data layout of a MEMS device.

the positioning time of the sled using a differential equation:

$$m\,\ddot{x} + \lambda\,\dot{x} + kx = Ku(t) \tag{1}$$

Equation 1 allows us, in particular, to model the spring like behavior exhibited when positioning the media sled; forces such as non-constant acceleration and damping not accounted for in related work [2, 3, 10]. Data attained from simulation has enabled us to observe equivalence regions, or areas of media that share the same seek times from a fixed point, where like data can be stored to improve access times. This technique of identifying equivalence regions for improving system performance through intelligent layout and scheduling has previously been applied to disk drives [6]. We present results that are contradictory to those presented in the MEMS literature, and propose an enhanced data layout technique based on these seek characteristics.

2 MEMS Background

In this section we provide a brief low-level description of a MEMS-based storage device. It is important to note that because this device is still in its infancy, many of the details are still uncertain. Although there are many potential architectures [5,12], we have based the physical parameters of our experimental model on the specification from Carnegie Mellon University (CMU) [1,2].

A MEMS-based storage is comprised of two main components: groups of probe tips called *tip arrays* that are used to access data on a movable *media sled* that slides in the x and y directions. Figure 1 illustrates the low level data layout of 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 2000 by 2000 bits in size. Multiple probe tips[1] read or write bits in parallel to their respective tip regions using orthogonal magnetic recording techniques. Each tip region is then separated into *tip tracks*, or the full stride of single tip. The tip tracks are further separated into *tip sectors*, the smallest accessible unit, analogous to disk drive sectors. Tip sectors are indexed by the tuple $\langle x,\ y,\ tip\rangle$, where x and y are some distance coordinates and *tip* is a tip number. Therefore, we define a tip track as all tip sectors who share the same x and *tip*, and a *cylinder* as all tip tracks who share the same x. To access data, the sled is positioned over the appropriate sectors and accessed in parallel by moving in the y direction.

[1]Because of power and heat considerations, not all tips will be able to be active at the same time. Instead, 200 to 2000 simultaneously active tips are expected.

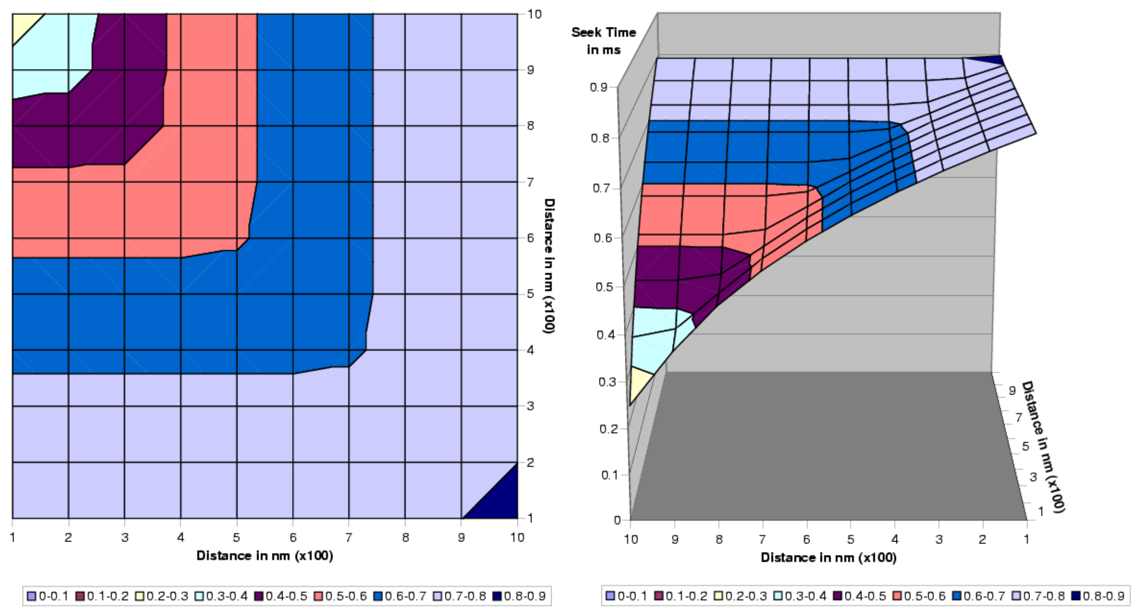

Figure 2: Equivalence regions for the top-left sector.

3 Seek Time Analysis

To develop an understanding of the seek time characteristics of a MEMS-based device, we conducted experiments using the UCSC positioning model and the physical parameters provided by CMU. By fixing a starting sector, we can then determine the physical distance and seek time to all other sectors on the device. We chose to examine three regions of the media sled in particular: the top-left, the center-left, and near-center. Because the device is symmetrical in nature, these positions map to all other corners of the device.

Figure 2 illustrates the seek time verses distance results gathered by setting the fixed sector to the top-left corner, and then seeking to all other sectors on the device. The various shades and, in the case of the 3D graph, levels, represent equivalent seek time regions based on a 0.1 ms granularity. There is a clear rectangular stratification, where layers of equivalent seek time regions exist on the media sled. Intuitively, seek times increase as the destination sector moves away from the starting sector. It is interesting to note that the regions grow thicker proportional to the distance away from the starting sector.

Figure 3 shows the surface and 3D seek times to all sectors, starting from the center-left sector. This data shows the same stratification as seen in Figure 2. What is more apparent in the surface graph, however, is the very rectangular relationship the equivalence regions have to the starting sector. Said another way, there exists a ratio of width and height where data benefits from being placed within that region. Figure 4 is a near-center representation of the same seek time analysis, and again shows the very rectangular nature of the equivalence regions.

These results are contradictory to those in the literature, as the device does not exhibit any benefit from a “cylindrical” layout, compared to modern disk drives. Placing data in cylinders will incur greater access latency contrasted to data that used a rectangular placement policy. For example, consider data written to a MEMS-device that exceeds a single cylinder in size. If we lay these data out within a single equivalence zone, then during random access, there will be no seeks worse than the cost that bounds that zone. Whereas a cylindrical layout will incur the seek the cost of crossing many equivalence zones, increasing access latency.

4 Future Work

Based on the results from Section 3, it is evident that a direct application of current disk drive data layout will be insufficient, and possibly detrimental to

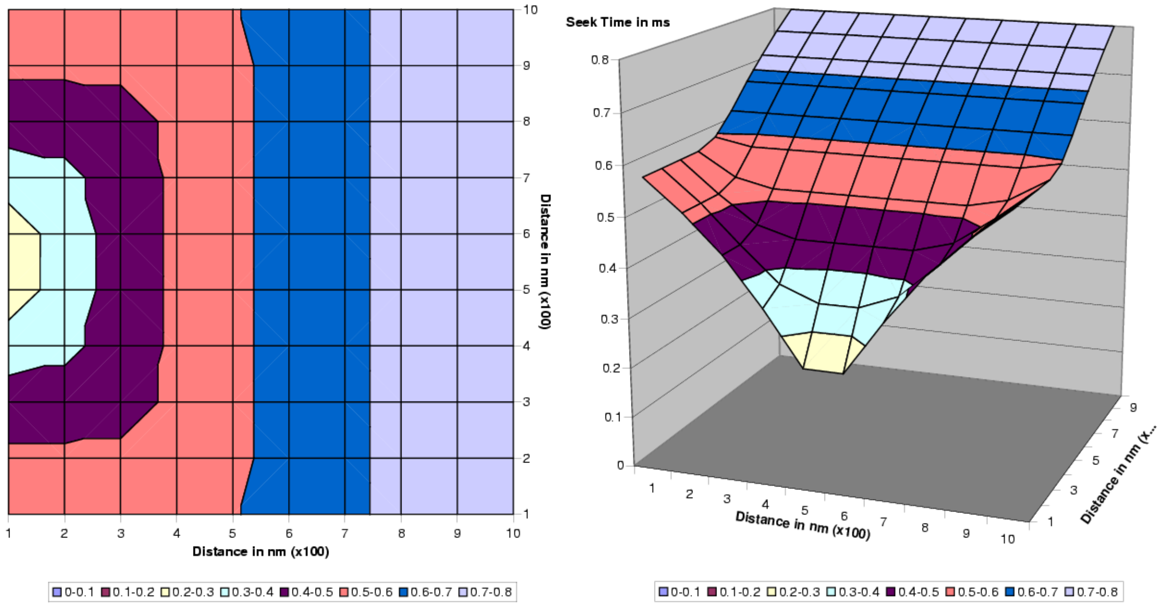

Figure 3: Equivalence regions for the center-left sector.

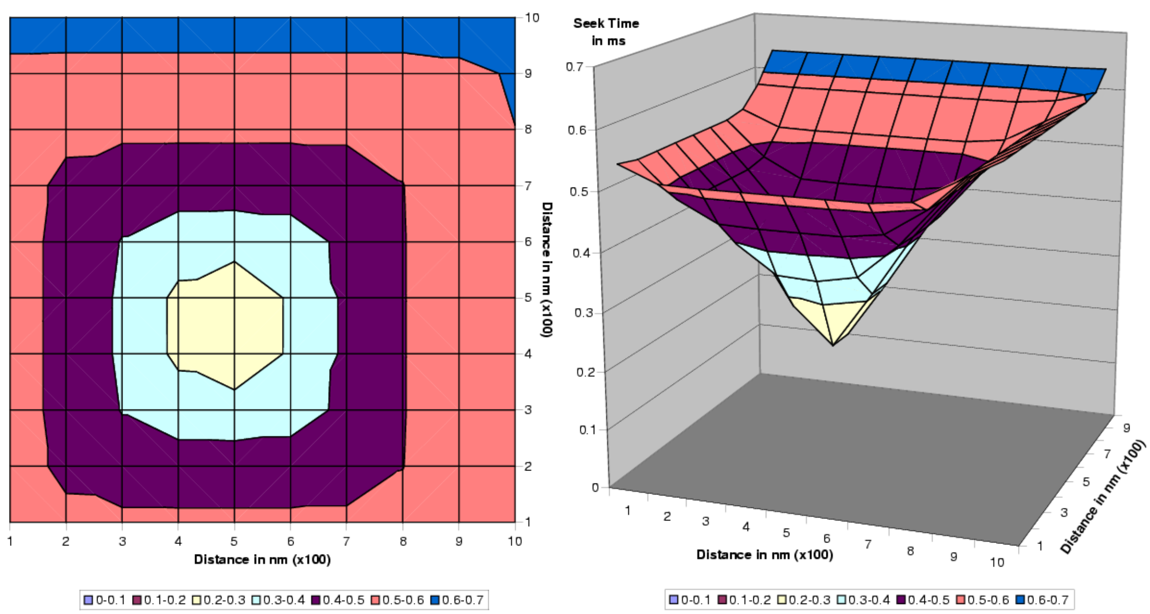

Figure 4: Equivalence regions for a near-center sector.

the performance of a MEMS-based storage device. It is our affinity to suggest a data layout policy that prefers rectangular placement over track and cylinder placement. By developing a heuristic for the cost of placing data in the x direction versus the y, we believe we can achieve greater performance from this two dimensional device. However, in order to further back our claim of rectangular stratification, we will need to perform similar seek time analysis on future generations of MEMS-device models as more information about the actual device characteristics are revealed. Furthermore, results comparing a ratio-based placement algorithm to the standard layout techniques will be necessary.

Acknowledgments

Thank you to Feng Weng for his help in understanding the MEMS-device simulator and to the other members of the CSL MEMS team: Ethan Miller and Lin Ying. Also thanks to the CMU PDL who have been gracious in sharing their data and hardware specifications with us.

References

[1] L. R. Carley, Gregory Ganger, and David Nagle. Designing computer systems with MEMS-based storage. *Communications of the ACM*, 43(11), November 2000.

[2] John L. Griffin, Steven W. Schlosser, Gregory R. Ganger, and David F. Nagle. Modeling and performance of MEMS-based storage devices. In *Proceedings of the 2000 SIGMETRICS Conference*, pages 56–65, June 2000.

[3] John L. Griffin, Steven W. Schlosser, Gregory R. Ganger, and David 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, October 2000.

[4] E. Grochowski. IBM leadership in disk storage technology.

[5] Nanochip Inc. Nanochip: Array nanoprobe mass storage IC. http://www.nanochip.com, 1999.

[6] David M. Jacobson and John Wilkes. Disk scheduling algorithms based on rotational position. Technical Report HPL-CSP-91-7rev1, Hewlett-Packard Laboratories, Concurrent Systems Project, March 1992.

[7] Tara Madhyastha and Katherine Pu Yang. Physical modeling of probe-based storage. In *Proceedings of the 18th IEEE Symposium on Mass Storage Systems and Technologies*, pages 207–224, April 2001.

[8] J. K. Peacock. The counterpoint fast file system. In *Proceedings of the USENIX Winter 1988 Technical Conference*, pages 243–250, Berkeley, CA, 1988. USENIX Association.

[9] Mendel Rosenblum and John K. Ousterhout. The design and implementation of a log-structured file system. *ACM Transactions on Computer Systems*, 10(1):26–52, February 1992.

[10] S. Schlosser, J. Griffin, Nagle, and G. Ganger. Designing computer systems with MEMS-based storage. In *Proceedings of the 9th Interational Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 1–12, 2000.

[11] Elizabeth Shriver, Eran Gabber, and Lan Huang Christopher A. Stein. Storage management for web proxies. In *Proceedings of the 2001 USENIX Annual Technical Conference*, pages 203–216, June 2001.

[12] 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.

[13] Bruce Worthington, Gregory Ganger, and Yale Patt. Scheduling algorithms for modern disk drives. In *Proceedings of the 1994 SIGMETRICS Conference*, pages 241–251, May 1994.