Techniques Mitigating Update-Induced Latency Overhead in Shingled Magnetic Recording

Kalyana Sundaram Venkataraman, Guiqiang Dong, and Tong Zhang

Department of ECSE, Rensselaer Polytechnic Institute, Troy, NY 12180 USA

Shingled writing has recently emerged as a promising candidate to sustain the historical growth of magnetic recording storage areal density. However, since the convenient update-in-place feature is no longer available in shingled recording, in order to update one sector, many sectors must be read and written back, leading to a significant update-induced latency overhead. This work develops two simple design techniques that can reduce such a latency overhead. Because the spatial locality of update-invoked read operations naturally enables the use of the 2-D read channel signal processing, the first technique aims to reduce update-invoked read latency by trading the SNR gain obtained by a 2-D read channel for higher disk rotation speed. Since update-induced latency overhead strongly depends on the location of the sectors being updated within each shingled region, the second technique aims to reduce the latency overhead by leveraging the data access locality in most real-time workloads in order to determine appropriate data placement. Through extensive simulations, we show that disk rotation speed boost assisted by a 2-D read channel can reduce the update latency by up to 33%, and data access characteristic sector placement can reduce the update latency by over one order of magnitude.

Index Terms—**Read channel, sector placement, shingled magnetic recording, update latency.**

I. INTRODUCTION

IN order to sustain the historical scaling of magnetic recording storage areal density, researchers have been pursuing several possible technologies, including heat-assisted magnetic recording, bit-patterned media, and shingled recording [1]. Different from the other alternatives, shingled recording [2], [3] uses the conventional head and media for recording, hence is an appealing short-term choice. Shingled recording relies on well-controlled track overlap to increase storage areal density, which nevertheless results in two issues. First, shingled recording apparently is subject to more significant inter-track interference (ITI). Although a two-dimensional (2-D) read channel can effectively compensate severe ITI, it could result in higher costs when using an array of read heads [4], [5]. Therefore, this work is only interested in shingled recording with a single read head, where conventional one-dimensional (1-D) read channel is sufficient to achieve the desired retry rate in normal random read. The second critical issue is the high latency overhead induced by update operations. In shingled recording, a certain number of tracks form a shingled region, and all the tracks within the same region must be written sequentially from one side to the other side. As a result, the convenient update-in-place feature is lost, i.e., when one sector is to be updated, instead of simply overwriting this sector, we have to read and write-back several adjacent tracks. This leads to a significant update-induced latency overhead.

This work is interested in reducing the update-induced latency overhead in shingled recording. Update-induced latency is primarily determined by two factors: the disk rotation speed and the physical location of the sector being updated within the shingled region. Accordingly, we have developed different techniques that can reduce this latency overhead by manipulating these two factors. First, we note that, during update operations, several adjacent tracks are read out consecutively, which makes ITI information explicitly available. Therefore, we can naturally apply 2-D read channel signal processing to realize certain signal-to-noise ratio (SNR) gain over 1-D read channel. Intuitively, we can trade the SNR gain to boost disk rotation speed. Conventionally, read heads should be aligned to the center of the target track, and 2-D read channel signal processing demands the readback signal from tracks on both sides. As a result, the read channel must incorporate a large buffer and read one more track. To address this issue, we propose to intentionally introduce a read head position offset in order to enable a one-sided 2-D read channel signal processing, which can achieve reasonably good performance with information from tracks only at one side. Finally, we note that most real-world workloads exhibit noticeable data access temporal and spatial locality. Since update-induced latency strongly depends on the physical location of the sectors being updated, we propose a simple scheme that can reduce update-induced latency by appropriately adjusting the physical location of sectors within each shingled region according to their data access characteristics.

We carried out simulations to demonstrate the effectiveness of the proposed design techniques. Based upon shingled recording read channel models presented in the open literature, we evaluated and compared the performance of various 1-D and 2-D read channel signal processing under different disk rotation speed. We also investigated the effects of different read head offset configurations on the performance of one-sided 2-D read channel. Results show that 2-D read channel assisted disk rotation speed boosting can reduce update-induced read latency by up to 33%. We further carried out trace-based simulations to evaluate the effectiveness of the data access characteristic aware sector placement, and the results show that it can reduce update-induced latency by over one order of magnitude.

Manuscript received July 28, 2011; revised October 05, 2011; accepted November 22, 2011. Date of publication December 07, 2011; date of current version April 25, 2012. Corresponding author: K. S. Venkataraman (email: venkak2@rpi.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMAG.2011.2178099

0018-9464/$26.00 © 2011 IEEE

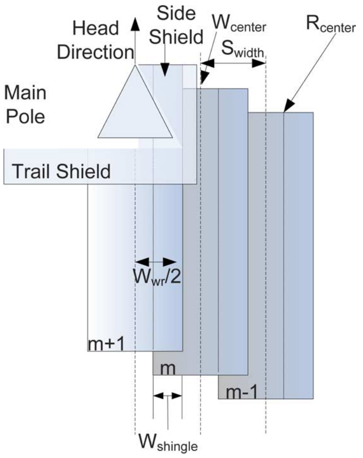


Fig. 1. Illustration of shingled magnetic recording with corner writer.

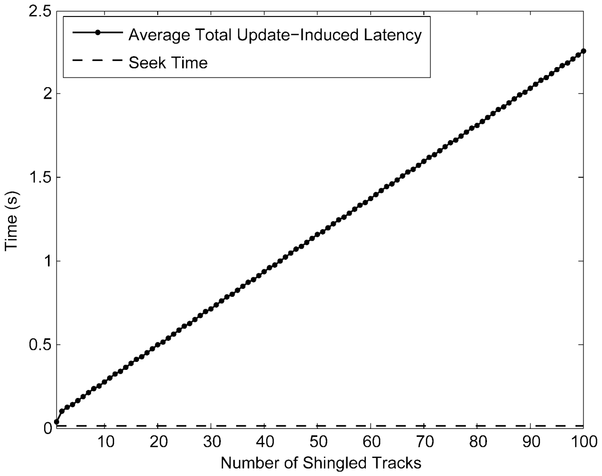


Fig. 2. Time versus number of shingled tracks for an update using a 3.5-in, 5400 rpm drive.

II. Background

Shingled magnetic recording works on the principle of overlapped writing and requires a corner writer to achieve high fields [6]. Fig. 1 illustrates its basic principle, where W_{Shingle} denotes the adjacent track overlap width, W_{center} and R_{center} are the track centers for writing and reading, and S_{width} denotes the shingle width. Although true two-dimensional processing enabled by multiple read heads can best mitigate the severe ITI in shingled magnetic recording [6]–[8], it apparently suffers from higher cost. Hence, this work only considers the use of a single read head.

Since shingled writing partially overwrites the adjacent track, the convenient update-in-place feature of conventional disk drives is lost. Thus, each update operation involves reading and writing many sectors and hence results in a significant latency overhead. Fig. 2 further demonstrates the significance of update-induced latency overhead and shows the typical update time for a 3.5-in, 5400 rpm drive with average seek time of 11 ms, where the number of shingled tracks being read ranges from 1 to 100. We set 4 kB sector size, a mean read data rate of 110 MB/s, 100 servo sectors, and assume negligible command processing time.

III. Proposed Design Techniques

Let m denote the number of shingled tracks within each shingled region. Dependent upon the location of the sector being updated, each update operation may read/write different number of tracks, ranging from 1 to m, leading to a different update-induced latency overhead. Let P_i denote the probability that one update operation involves i shingled tracks, τ_{rot} denote the latency of one disk rotation, and τ_{seek} denote the read head seek latency, we can express the average total update-induced latency (including read and write latency) as

$$\tau_{\text{seek}} + 2\tau_{\text{rot}} \cdot \sum_{i=1}^{m}(P_i \cdot i). \tag{1}$$

This clearly suggests that, in order to reduce the update-induced latency overhead, we have primarily two options: (i) increase the disk rotation speed during update, (ii) increase the probability P_i for smaller i. Following this intuitive observation, we develop different techniques for reducing update-induced latency overhead as described in this section.

A. Disk Rotation Speed Boost

We first investigate the potential of reducing update-induced latency overhead by temporarily boosting the disk rotation speed. We note that, although variable rotation speed has been widely discussed in open literature [9]–[12], prior work has focused on improving hard disk drive energy efficiency. The practical feasibility of variable rotation speed has been demonstrated by commercial hard disk drives (e.g., variable speed drives from WD [13]).

Disk rotation speed is typically limited by the read channel, i.e., as we increase the disk rotation speed, the SNR of the readback signal will degrade, and the target disk retry probability will directly determine the worst-case signal SNR and hence the maximum-allowable disk rotation speed. Even if the disk retry probability is allowed to increase to a certain degree, the retry penalty could offset the latency reduction gained through a disk rotation speed boost. In addition, switching the disk rotation speed incurs a speed transition time penalty τ_{trans}, which can be tens or hundreds of ms. We define P_{retry} as the retry probability corresponding to an error event, i.e., in the event of an ECC decoding failure, the sector is re-read. Let $\tau_{\text{rot}}^{(\text{norm})}$ and $\tau_{\text{rot}}^{(\text{high})}$ denote the disk rotation latency under normal and boosted speed and define $L = \sum_{i=1}^{m}(P_i \cdot i)$. We assume that the disk retry probability under normal disk rotation speed is sufficiently low and can be ignored in the calculation. Hence, when using a disk rotation speed boost, we can express the update-induced latency overhead as

$$\begin{aligned}\tau_{\text{lat}} &= \tau_{\text{seek}} + \tau_{\text{rot}}^{(\text{high})} \cdot L + (1 - P_{\text{retry}}) \cdot \tau_{\text{rot}}^{(\text{high})} \cdot L \\ &\quad + P_{\text{retry}} \cdot 2\tau_{\text{rot}}^{(\text{norm})} \cdot L + \tau_{\text{trans}} \\ &= \tau_{\text{seek}} + \tau_{\text{trans}} + 2\tau_{\text{rot}}^{(\text{high})} \cdot L \\ &\quad + P_{\text{retry}} \cdot (2\tau_{\text{rot}}^{(\text{norm})} - \tau_{\text{rot}}^{(\text{high})}) \cdot L.\end{aligned} \tag{2}$$

Since an update operation reads consecutive tracks and ITI is the dominating noise source, it is indeed pragmatic to apply

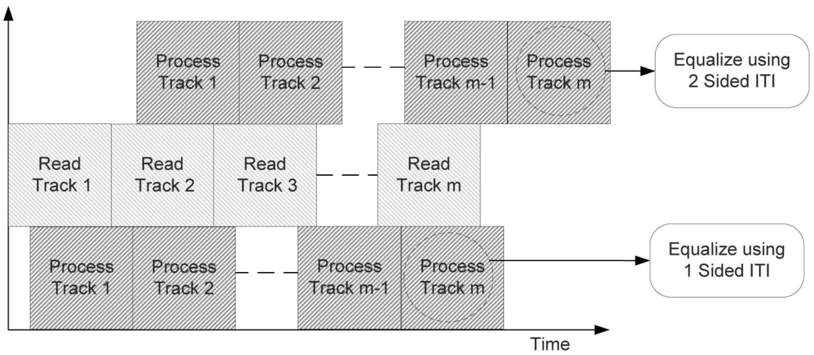


Fig. 3. Comparison between one-sided and two-sided 2-D read channel signal processing in terms of latency (note that the track 1 is on the edge of one shingled region, hence it only has ITI from one side).

a 2-D read signal processing to explicitly compensate for ITI during an update-induced read to reduce the retry probability P_{retry}. Therefore, it is natural to employ a *dual-mode* read channel: for a normal host-induced read access, the disk rotates at its normal speed and the read channel executes a 1-D signal processing by treating ITI as random noise; for an update-induced read access, the disk rotates with the boosted speed and the read channel executes a 2-D signal processing by explicitly incorporating readback signals of adjacent tracks. Assuming that the 2-D read channel signal processing only involves the two immediately adjacent tracks, we can express the overall update-induced latency overhead as

$$\begin{aligned}\tau_{\text{lat}}^{2-D} = \tau_{\text{seek}} + \tau_{\text{trans}} + \tau_{\text{rot}}^{(\text{high})} + 2\tau_{\text{rot}}^{(\text{high})} \cdot L \\ + P_{\text{retry}}^{2-D} \cdot \left(2\tau_{\text{rot}}^{(\text{norm})} - \tau_{\text{rot}}^{(\text{high})}\right) \cdot L \quad (3)\end{aligned}$$

where P_{retry}^{2-D} denotes the retry probabilities when using a 2-D read channel signal processing.

B. One-Sided 2-D Read Channel Signal Processing

The above design strategy employs a 2-D read channel signal processing to compensate for the SNR loss induced by a higher disk rotation speed. Straightforwardly, a 2-D read channel signal processing involves readback signals from tracks on both sides of the main track, which is assumed in the above discussion and referred to as two-sided 2-D read channel signal processing. Although it can achieve the best possible read channel signal processing performance, it comes with two types of penalties: (i) we have to read one more track during update-invoked read operations, leading to an extra latency overhead of one disk rotation, e.g., suppose one update request demands the read-and-write of sectors on s consecutive tracks, two-sided 2-D read channel signal processing must read the sectors on $s+1$ tracks; and (ii) the read channel must provide sufficient memory such that it can at least store the readback signals of sectors on three consecutive tracks, leading to a silicon cost overhead.

Intuitively, the above issues can be addressed if the 2-D read channel signal processing only involves readback signals on just one side of the main track, which is referred to as one-sided 2-D read channel signal processing. Assuming only the immediately adjacent tracks are involved in the 2-D read channel signal processing, Fig. 3 illustrates the comparison between the one-sided and two-sided 2-D read channel signal processing in terms of latency. Since the extra latency of an additional disk rotation becomes more significant as the number of tracks being read reduces, it is particularly desirable to use a one-sided over two-sided 2-D read channel signal processing when update operations involve only a few tracks. However, one-sided 2-D read channel signal processing clearly suffers from SNR loss compared with its two-sided counterpart, leading to a higher retry probability.

We propose a simple design technique to reduce the SNR loss incurred by the use of a one-sided 2-D read channel signal processing. In the current design practice, the read head is aligned with the center of the main track, leading to similar and minimal ITI from tracks on both sides, and noticeable read head position offset is considered as track mis-registration and should be avoided. In one-sided 2-D read channel signal processing, if we still keep the read head aligned with the center of the main track, almost half of total ITI due to the asymmetric writer in SMR cannot be explicitly compensated. We propose to intentionally offset the read head with respect to the main track center as shown in Fig. 4, where the H_{offset} denotes the intentional offset distance. Although the total ITI increases, it is relatively dominated by the ITI from the adjacent track that has already been read in the previous disk rotation, which can be explicitly compensated by a one-sided 2-D read channel signal processing. We note that a recent work [14] has employed an intentional read head off-set to improve the ITI cancellation effectiveness during random data accesses.

C. Access Characteristics Aware Sector Placement

When a sector is updated, the physical location of this sector within the shingled region determines the number of tracks that must be read and written, i.e., the physical location of the sector being updated directly determines the update-induced latency overhead. Without loss of generality, we assume that the data is recorded onto each shingled region from the inner most track to the outer most track. Therefore, sectors closer to the outer most track tend to induce less update latency overheads. This observation directly leads to a simple design principle: we should try to place those data, which will be more likely updated, closer to the outer most track within each shingled region, i.e., we should try to increase the probability P_i for smaller i as in (1). This is referred to as access characteristics aware sector placement.

The practical realization of this simple design principle involves two issues: (i) how to determine on-the-fly which sectors will be more (and less) likely updated, and (ii) when should

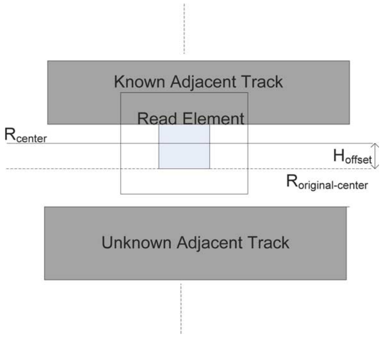


Fig. 4. Illustration of using intentional read head offset to reduce SNR loss of one-sided 2-D read channel signal processing.

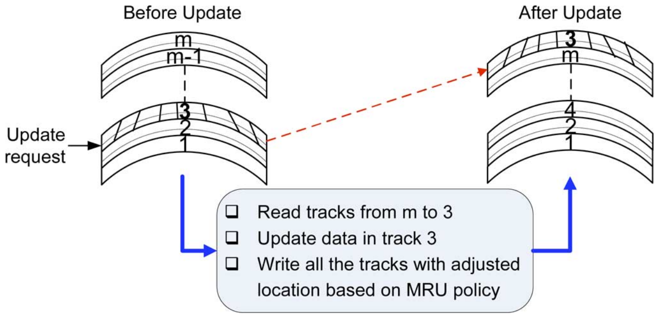


Fig. 5. Illustration of the access characteristics aware sector placement using the most recently used (MRU) policy.

we adjust the physical placement of sectors according to their future update likelihood. The first issue is the same as the one faced by the cache replacement problem, which has been widely studied (e.g., see [15]–[18]). Prior work has developed many different strategies that maintain a (partially) ordered list of entries on-the-fly based upon data access history and certain update rules. In this work, to simplify the implementation and hence reduce the system-level overhead, we propose to use the simplest most recently used (MRU) policy with the granularity of a track, i.e., we only keep the record of the most recently updated track, and try to move this track to the outer most location of the shingled region. Regarding the second issue above, we propose to adjust the physical placement of tracks only during the update operations in order to eliminate any extra overhead. During each update, after reading all the involved tracks, we adjust their physical placement according to their future update likelihood when writing them back. We further illustrate the proposed strategy using the following example: As illustrated in Fig. 5, assume one shingled region contains m tracks and a sector in the third track needs to be updated, we have to first read all the $m-2$ tracks (from the mth track to the third track), then update the sector in the 3rd track, and write all the $m-2$ tracks back by moving the most recently accessed track (i.e., the third track) to the outer most location.

IV. SIMULATION RESULTS

This section presents simulation results to demonstrate the effectiveness of our proposed design techniques. According to [19], [20], we set the impulse response of shingled magnetic recording channel as

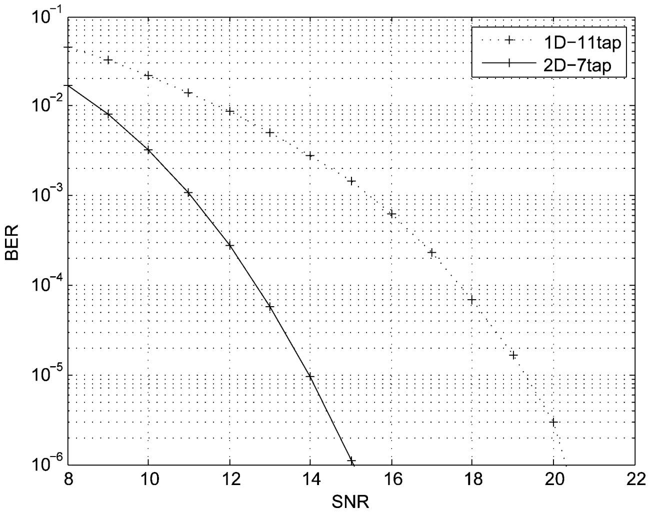


Fig. 6. Simulated BER results when using either a 7-tap 2-D equalization or a 11-tap 1-D equalization followed by a 1-D Viterbi detection.

$$h(t,\lambda)=\exp(-1.34898^2(t^2+\lambda^2)/2(\mathrm{PW}_{50}/T_b)^2) \qquad (4)$$

where t and λ represent the along-track and cross-track location, T_b is the bit width, and PW_{50} is the half power width. We consider the ideal read head element as specified by Wood [6] with sensor length L of 4 nm, sensor width W of 3 nm, and a magnetic separation of 2 nm. We set the adjacent track overlap width W_{Shingle} corresponds to a 20% ITI with the shingle width S_{width} set to 6.40 nm. The half power width along track PW_{50} is 5.2 nm with a bit separation of 5.2 nm along track. We set the rpm of 5400 with AWGN as the baseline scenario, with media jitter noise which follows a zero mean normal distribution corresponding to $\sigma_j=0.01$ [21]. As we increase the the disk rotation speed, the bit width T_b and jitter noise are linearly scaled based on the baseline.

In this work, the 2-D read channel signal processing is realized by a 2-D equalizer followed by a conventional 1-D Viterbi detection. Readback signals are equalized to either a 1-D or a 2-D generalized partial response (GPR) with a 1-D target estimated by a standard Lagrange constrained minimization [22], [23]. Fig. 6 compares the 1-D and the two-sided 2-D read channel bit error rate (BER) for a 5400 rpm drive with ITI component of around 20%, where the 2-D equalizer has 7 taps per track while the 1-D equalizer has 11 taps. It clearly demonstrates the advantage of using a 2-D read channel signal processing in shingled recording with significant ITI.

First, we investigate the effectiveness of boosting the disk rotation speed with the two-sided 2-D equalization. As observed in Fig. 6, the 2-D equalization can achieve a large SNR gain over its 1-D counterpart (i.e., around 4.5 dB at BER of 10^{-4} in this case study), which can lead to a large potential for increasing disk rotation speed during update-invoked read operations. Fig. 7 shows the simulation results when using a two-sided 2-D equalization under different disk rotation speeds. For the purpose of comparison, it also shows the results when using a

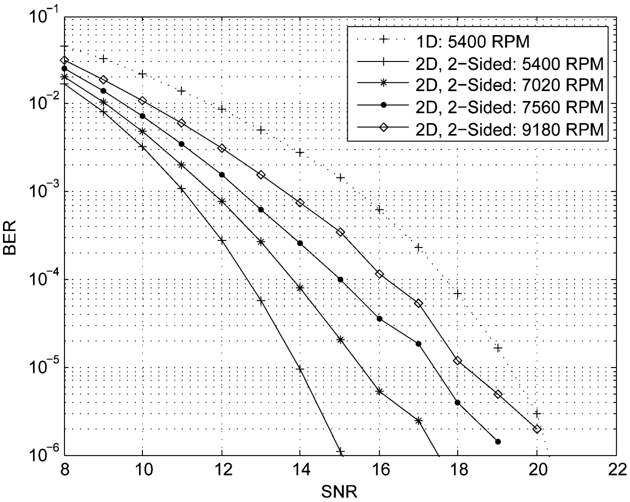


Fig. 7. Simulated BER results under different disk rotation speed when a two-sided 2-D equalization is used.

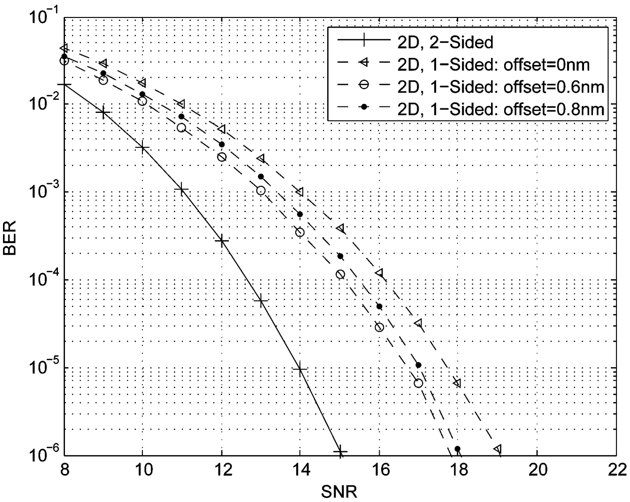


Fig. 8. Simulated BER results under different read head offset for a one-sided 2-D equalization.

1-D equalizer at 5400 rpm. The results show that the disk rotation speed can be boosted up to 9180 rpm from 5400 rpm.

Next, we evaluate the effectiveness of the proposed one-sided 2-D equalization with intentional read head offset. Fig. 8 shows the simulation results under different read head offset values. For the purpose of comparison, it also shows the simulation results when a two-sided 2-D equalization is being used. By having an offset of around 0.6 nm, which corresponds to around 10% of the bit dimension in this case study, we are able to reduce the SNR loss due to the use of one-sided 2-D equalization by around 1 dB. Fig. 9 shows the simulations results under different disk rotation speeds. For the purpose of comparison, it also shows the results when using a 1-D equalization under 5400 rpm. The results show that the disk rotation speed during the update operation can be increased up to 7560 rpm when a one-sided equalizer is used.

Fig. 10 shows the average update-induced latency overhead under different scenarios when varying the number of tracks within each shingled region, where we fix the BER as 10^{-6} for all the scenarios to ensure a fair comparison. We assume all the sectors have the same probability for update. We set the disk rotation speed transition penalty as 100 ms. We note that if the controller can group several update requests together, the speed transition penalty can be accordingly reduced. The results show that the speed boost is effective when the number of tracks within each shingled region is greater than 11 for two-sided read channel at 9180 rpm and greater than 15 for one-sided read channel at 7560 rpm. Finally, we note that when we use the rpm of 5400 with a 1-D read channel as the baseline, and assume that the BER performance of higher rotation speed with a 2-D read channel cannot be worse than that of the baseline. As a result, the simulations suggest that 9180 rpm with the two-sided 2-D read channel and 7560 rpm with the one-sided read channel can achieve BER close to the baseline. Hence, we stop at these rpm values in this case study.

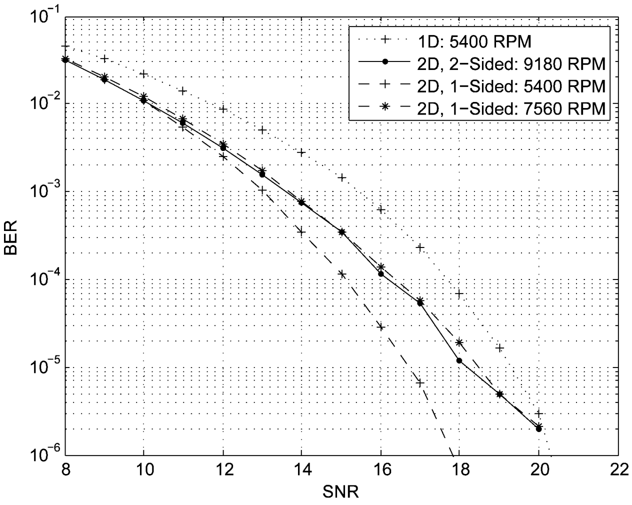
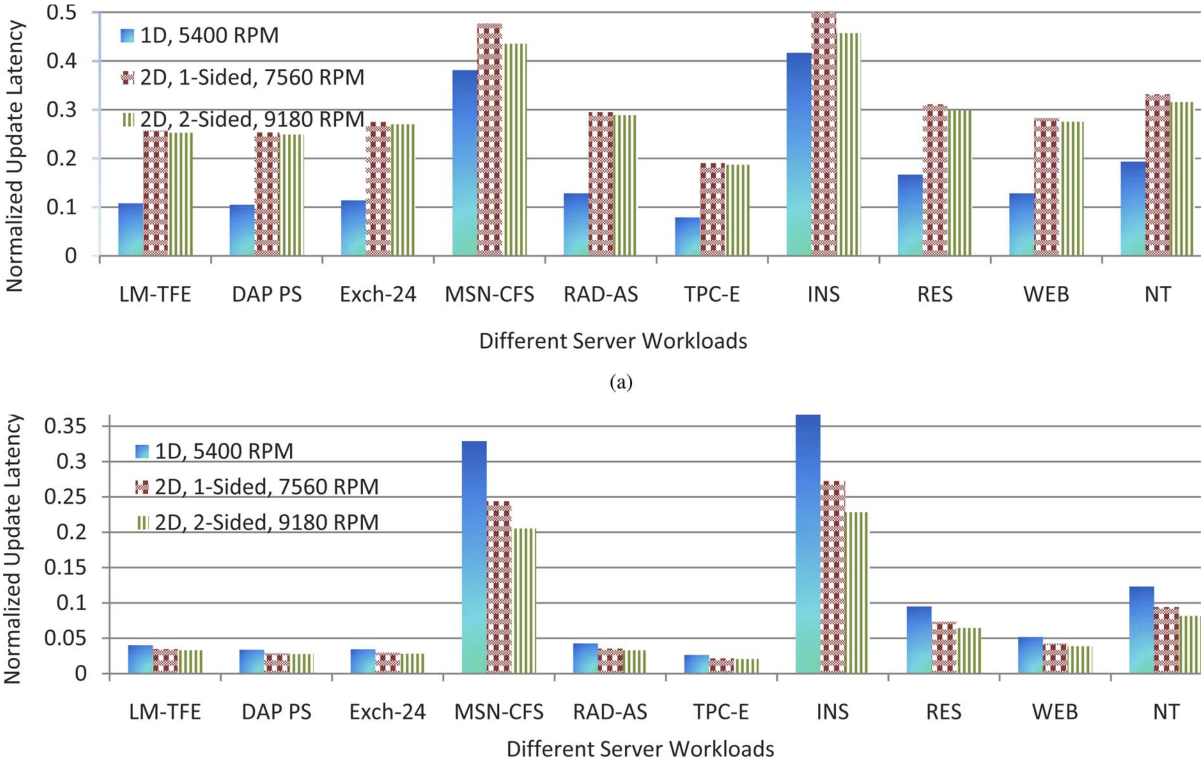


Fig. 9. Simulated BER results under different disk rotation speed when one-sided 2-D equalization is being used.

Finally, we evaluate the effectiveness of the access characteristics aware sector placement. As pointed out earlier, we use the simplest MRU policy to move the most recently updated track to the outer most position in each shingled region. We use various server workload traces including INS, RES, WEB, and NT traces from [24], and DAP-PS, Exch-24, LM-TFE, MSN-CFS and RAD-AS traces from [25]. We set the boosted disk rotation speed during update-invoked read operations as 7560 rpm and 9180 rpm in the case of one-sided and two-sided 2-D equalization, respectively. We set the disk rotation speed transition penalty of 100 ms. In the simulations, we considered four different scenarios: (i) 1-D read channel at 5400 rpm without MRU, (ii) 1-D read channel at 5400 rpm with MRU, (iii) one-sided 2-D read channel at 7560 rpm with MRU, and (iv) two-sided 2-D read channel at 9180 rpm with MRU. By normalizing the update-induced latency overhead based upon the first scenario, Fig. 11 shows the normalized latency overhead for different workloads when each shingled region contains 25 tracks and 100 tracks. The results clearly show that the access characteristics aware sector placement strategy can effectively reduce the update-induced read latency overhead over a wide range of workloads. Intuitively, workloads with higher write access variations (i.e., a small percentage of data are updated much more frequently than the others) tend to benefit more from this strategy. For example, the LM-TFE



(a)

(b)

Fig. 10. Average update-induced latency variation under different disk rotational speeds when varying the number of tracks within each shingled region.

workload trace represents the workload of live map front-end servers hosting map images, for which the hot-data accounts for only 2% of total data. Hence, the corresponding latency reduction is significant as shown in the figures. Other workload traces with similar characteristics include DAP-PS (for advertisement display servers), Exch-24 (for exchange servers), RAD-AS (for authentication servers for wireless access), and WEB (for typical web servers). On the other hand, this design strategy tends to be less effective for those workloads with less write access variations, e.g., the MSN-CFS trace for metadata storage and INS trace for instruction workload servers.

The simulation results show that the size of shingled region can largely affect the effectiveness of the proposed design techniques. If the number of tracks per shingled region is small, one update operation may only involve very few tracks when using access characteristics aware sector placement. As a result, the gain obtained from disk rotation speed boost may not be able to offset the disk rotation speed transition penalty. As shown in Fig. 11(a), when each shingled region only contains 25 tracks, disk rotation speed boost does not bring any gain compared with using normal speed. As we increase the size of shingled region, more and more tracks will be involved in each update operation even when using access characteristics aware sector placement. Hence, the use of disk rotation speed boost can be beneficial, as shown in Fig. 11(b) for 100 tracks per shingled region.

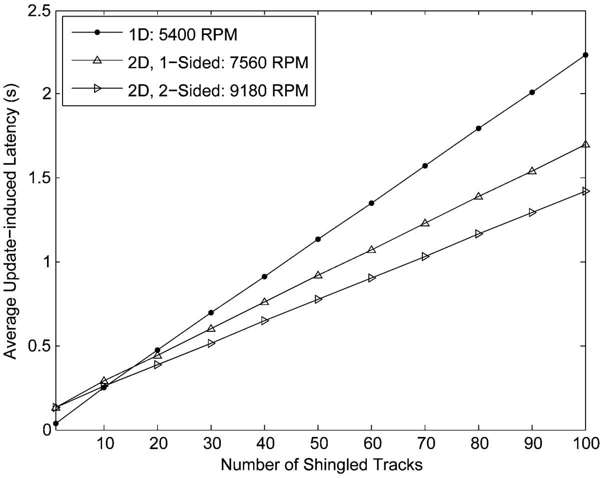


Fig. 11. Comparison of update latencies for different workload traces with MRU, normalized by the latency when using 1-D read channel without MRU at 5400 rpm. (a) Each shingled region contains 25 tracks. (b) Each shingled region contains 100 tracks.

V. CONCLUSION

This paper presents simple design techniques that can reduce update-induced latency overhead in shingled recording. Motivated by the fact that update-invoked read operations naturally enables the use of the 2-D read channel signal processing, we first propose to trade the SNR gain obtained by a 2-D read channel signal processing for increasing disk rotation speed during update and hence reducing its latency overhead. We further propose to use a one-sided 2-D read channel with intentional read head offset in order to reduce the silicon cost. We also propose a simple technique that exploits data access locality in most workloads to reduce update-induced latency overhead. The effectiveness of these techniques has been well demonstrated through simulations.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under Grant ECCS-1128148.

REFERENCES

[1] Y. Shiroishi *et al.*, “Future options for HDD storage,” *IEEE Trans. Magn.*, vol. 45, no. 2, pp. 917–923, Feb. 2009.

[2] K. Miura, E. Yamamoto, H. Aoi, and H. Muraoka, “Estimation of maximum track density in shingles writing,” *IEEE Trans. Magn.*, vol. 45, no. 10, pp. 3722–3725, Oct. 2009.

[3] F. Lim, B. Wilson, and R. Wood, “Analysis of shingle-write readback using magnetic-force microscopy,” *IEEE Trans. Magn.*, vol. 46, no. 6, pp. 1548–1551, Jun. 2010.

[4] L. Barbosa, “Simulataneous detection of readback signals from magnetic recording tracks using array heads,” *IEEE Trans. Magn.*, vol. 26, no. 5, pp. 2163–2165, Sep. 1990.

[5] P. Voois and J. Cioffi, “Multichannel signal processing for multiple-head digital magnetic recording,” *IEEE Trans. Magn.*, vol. 30, no. 6, pp. 5100–5114, Nov. 1994.

[6] R. Wood, M. Williams, A. Kavčić, and J. Miles, “The feasibility of magnetic recording at 10 terabits per square inch on conventional media,” *IEEE Trans. Magn.*, vol. 45, no. 2, pp. 917–923, Feb. 2009.

[7] K. S. Chan *et al.*, “Channel models and detectors for two-dimensional magnetic recording,” *IEEE Trans. Magn.*, vol. 46, no. 3, pp. 804–811, Mar. 2010.

[8] A. Kavčić, X. Huang, B. Vasic, W. Ryan, and M. F. Erden, “Channel modeling and capacity bounds for two-dimensional magnetic recording,” *IEEE Trans. Magn.*, vol. 46, no. 3, pp. 812–818, Mar. 2010.

[9] H. Yada *et al.*, “Head positioning servo and data channel for HDD’s with multiple spindle speeds,” *IEEE Trans. Magn.*, vol. 36, no. 5, pp. 2213–2215, Sep. 2000.

[10] K. Okada, N. Kojima, and K. Yamashita, “A novel drive architecture of HDD: Multimode hard disk drive,” in *Proc. Consumer Electronics, (ICCE)*, Los Angeles, CA, Jun. 2000, pp. 92–93.

[11] S. Gurumurthi, A. Sivasubramaniam, M. Kandemir, and H. Franke, “Reducing disk power consumption in servers with DRPM,” *IEEE Computer Society*, vol. 36, no. 12, pp. 59–66, Dec. 2003.

[12] S. W. Son and M. Kandemir, “Energy-aware data prefetching for multispeed disks,” in *Proc. 3rd ACM Conf. Computing Frontiers*, Ischia, Italy, May 2006, pp. 105–114.

[13] Western Digital [Online]. Available: http://www.wdc.com

[14] E. F. Haratsch, G. Mathew, J. Park, M. Jin, K. J. Worrell, and Y. X. Lee, “Intertrack interference cancellation for shingled magnetic recording,” *IEEE Trans. Magn.*, vol. 47, no. 10, pp. 3698–3703, Oct. 2011.

[15] K. So and R. N. Rechtschaffen, “Cache operations by MRU change,” *IEEE Trans. Comput.*, vol. 37, no. 6, pp. 700–709, Jun. 1988.

[16] B. Ozden, R. Rastogi, and A. Silberschatz, “Buffer replacement algorithms for multimedia storage systems,” in *Proc. IEEE Multimedia*, 1996, pp. 172–180.

[17] M. Kandemir, F. Li, M. J. Irwin, and S. W. Son, “A novel migration based NUCA design for chip multiprocessors,” in *Proc. IEEE/ACM Conf. High Performance Computing, Networking, Storage and Analysis*, Austin, TX, Aug. 2008.

[18] J. Jeong and M. Dubois, “Cache replacement algorithms with nonuniform miss costs,” *IEEE Trans. Comput.*, vol. 55, no. 4, pp. 353–365, Apr. 2006.

[19] D. T. Wilton, D. M. McKirdy, H. A. Shute, J. J. Miles, and D. J. Mapps, “Approximate three-dimensional head fields for perpendicular magnetic recording,” *IEEE Trans. Magn.*, vol. 40, no. 1, pp. 148–156, Jan. 2004.

[20] E. Hwang, R. Negi, and B. V. K. V. Kumar, “Signal processing for near 10 Tbit/in^2 density in two-dimensional magnetic recording,” *IEEE Trans. Magn.*, vol. 46, no. 6, pp. 1813–1816, Jun. 2010.

[21] J. Moon, “Signal-to-noise ratio definition for magnetic recording channels with transition noise,” *IEEE Trans. Magn.*, vol. 36, no. 5, pp. 3881–3883, Sep. 2000.

[22] S. Nabavi and B. V. K. V. Kumar, “Two-dimensional generalized partial response equalizer for bit-patterened media,” in *Proc. IEEE-ICC*, Glasgow, Scotland, Jun. 2007, pp. 6249–6252.

[23] W. Chang and J. R. Cruz, “Inter-track interference mitigation for bit-patterened magnetic recording,” *IEEE Trans. Magn.*, vol. 46, no. 11, pp. 3899–3908, Nov. 2010.

[24] D. Roselli, J. R. Lorch, and T. E. Anderson, “A comparison of file system workloads,” in *Proc. USENIX Annual Technical Conf.*, San Diego, CA, 2000, pp. 41–54.

[25] S. Kavalanekar, B. Worthington, Q. Zhang, and V. Sharda, “Characterization of storage workload traces from production windows servers,” in *Proc. IEEE Int. Symp. Workload Characterization (IISWC)*, Austin, TX, Oct. 2008, pp. 119–128.