Power Conscious Disk Scheduling for Multimedia Data Retrieval*

Jungwan Choi[1], Youjip Won[1], and S.W. Nam[2]

[1] Division of Electrical and Computer Engineering, Hanyang University,
17 Hangdangdong Seongdongku, Seoul, Korea
{chrys|yjwon}@ece.hanyang.ac.kr
[2] swnam@hanyang.ac.kr

Abstract. In this paper, we present the novel algorithm called *Adaptive Round Merge(ARM)* of retrieving the multimedia data blocks for real-time playback. We focus on minimizing the power consumption involved in multimedia data retrieval. Different from the legacy text based data, real-time multimedia playback requires that the storage supplies the data block *continuously* until the end of the playback. This puts immense burden on the power scare environment since the disk has to be *active* for the entire playback duration. Our algorithm, ARM, carefully analyzes the power consumption profile of the disk drive and establishes the data retrieval schedule for the given playback. The objective is to minimize the power consumption. It determines the amount of data blocks to read, the length of *active* and *idle* period. According to our simulation result, the ARM algorithm reduces the power consumption by as much as 40% compared to the *Full Buffering* strategy. It manifests itself when the playback length is relatively short, typically less than 30 sec.

Keywords: Disk Scheduling, Power Management, Multimedia, Playback, Mobile Device

1 Introduction

1.1 Motivation

Electricity is the prime commodity in modern mobile devices, e.g. smart phone, PDA, MP3 player and etc. Due to the recent rapid deployment of the mobile devices and the concern of environmental impact of electronic systems, the low power design is getting emphasized more and more. Among the numerous hardware components in the modern computer system, the disk drive is one of the major devices where the power consumption takes up a significant fraction of entire system. The disk portion of the power consumption has decreased in past few years from 25% to 10% [14]. While the disk based storage device, e.g. hard

* This work is funded by KOSEF through Statistical Research Center at Seoul National University.

T. Yakhno (Ed.): ADVIS 2002, LNCS 2457, pp. 336–345, 2002.
© Springer-Verlag Berlin Heidelberg 2002

disk and optical disk becomes small enough to be used in mobile devices, the practical usage of which leaves much to be desired due to the stringent power consumption restriction of the mobile device.

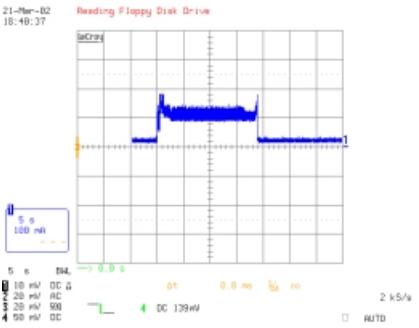
The typical approach to reduce the power consumption for hard disk drive is to shutdown the disk when it is not in use [9]. Disk shutdown algorithms can be classified into two categories: *predictive* [5] and *stochastic schemes* [4]. Although stopping the plate spinnig can contribute to reducing the power consumption, it consumes extra energy in spinning up the platter. Thus, it is important to elaborately model the energy consumption profile for a given workload and to establish the disk operation schedule. When the digital device is used for dedicated purpose, its storage access characteristics can be rather simple and may be easier to predict. MP3 player, digital camera, or Personal Video Recoder can be typical examples. When the target application of the device is clearly defined, we believe that it is possible to achieve greater reduction in power consumption by effectively exploiting the workload characteristics and by incorporating it into I/O scheduling.

In this work, we'd like to address the problem of minimizing the power consumption of the disk drive for real-time multimedia data playback. The major issue in supporting the real-time playback of multimedia data in the local storage is *how to guarantee the continuous flow of data.* Round based disk scheduling algorithm has been widely used to support the continuous flow of the data [7, 11]. Preceding works assume that the disk drive operates in the *steady* state and thus the maximum transfer rate of the disk does not change. However, disk drive controlled by dynamic power management can adaptively change the state of operation, e.g. *active* or *idle*, with respect to the state of workload [12,14]. While this feature can significantly extend the battery life, it adds another dimension of complexity in scheduling of the multimedia data retrieval. We elaborately model the power consumption behavior of the low power disk drive and develop an *Adaptive Round Merge(ARM)* scheduling algorithm which guarantees real-time multimedia playback while minimizing the power consumption involved in operating the storage device. We presently assume that the multimedia file is constant bit rate(CBR) encoded.

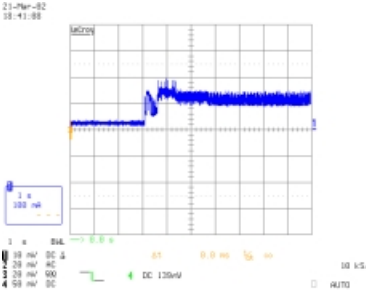
1.2 Related Works

Power management on mobile devices attracts significant interests in the academia as well as in the industry and there have been a number of works regarding the hard disk power management. A set of hard disk requests are grouped into a number of distinct sessions, and disk is shutdown between the sessions the prime issue is how to detect the session termination. Simunic et al. [12] used semi-Markov decision process model. Lu et al. [9] adaptively changes the threshold value for detecting session termination. By increasing the number of disk states, it is possible to reduce the power consumption of the disk [14]. Greenawalt [6] formulates the relationship between the number of disk state transitions, power consumption, and system performance. There have been a

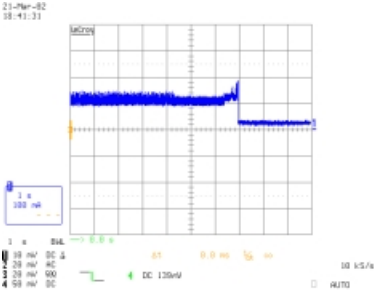
number of power management approach from the operating system's point of view [3,10,8,13].



(a) Power Consumption Profile for entire playback duration



(b) Power Consumption profile during Startup phase



(c) Power Consumption profile during Spindown phase

Fig. 1. Power Consumption Profile of IBM MicroDrive(DMDM-10340)

The above mentioned power management strategy assumes that the device operates in the *generic* workload. When the device(or embedded system) is designed to perform specific task, e.g. MP-3 player, digital camera, personal video recorder, etc., it is possible to further effectively exploit the characteristics of the underlying workload to minimize the power consumption of the storage drive.

2 Power Consumption Profile of Low Power Disk Drive

We first examine the power consumption profile of the low power disk drive[1]. Fig. 1 illustrates the power consumption profile of the lower power disk drive. We

[1] It has been measured using WavePro 950 from LeCroy. Sampling rate is 16 Gigasamples/sec

measure the power consumption in real-time playback of multimedia data in the IBM MicroDrive(DMDM-10340). Playback rate and the playback duration are 1.5 MBits/sec and 20 sec, respectively. Fig. 1(a) shows the power consumption during entire playback. Fig. 1(b) magnifies the power consumption behavior during the start of the playback. We can find that spinning up the disk requires extra overhead in power consumption and it takes 1.7 sec before the disk reaches the steady state. Fig. 1(c) illustrates the power consumption profile during the disk shutdown. This happens because there is no more requests. In Fig. 1(c), it takes about 0.8 sec for the disk to change into the idle state. This type of power consumption characteristics applies to the most disk drives including CD-ROM drive, DVD, hard disk, optical disk, etc.

We model the power consumption behavior of the low power disk drive as in Fig. 2. Retrieving data blocks from the disk drive consists of three phases: *Startup*, *Read*, and *Finish* phases. The device is initially in *idle* state. Startup phase includes the operation of *spin up*, *focus* and *tracking*. In *Read* phase, the disk drive transfers data. When there is enough buffer space in memory, the disk drive retrieves the data block at its maximum transfer rate. When the buffer is full, the speed of retrieval is bounded by the data consumption rate. In case of multimedia playback, it corresponds to the playback rate of the file. When I/O request queue becomes empty, the disk drive *finishes* its operation and goes into *idle* phase. *Finish* phase includes *spin down* and head parking. It is important to note that whether the storage media is detachable or not, commodity small size storage devices retrieve the data based on *startup*, *read*, and *finish* phases. For real-time multimedia playback, an application issues a burst of reads pe-

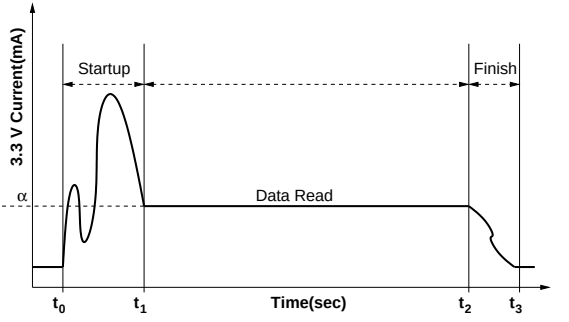


Fig. 2. Schematic View of Power Consumption Profile of Data Read Operation

riodically and the disk supplies the data block in periodic fashion conformant to the playback rate. The application program in this context can be the VoD server or the MPEG player dependent upon whether data is located in remote or local storage. The size of a read burst and the interval between the adjacent read bursts needs to be carefully determined based upon the available buffer size, tolerable startup latency, disk overhead such as seek and rotational latency [15]. The low power disk has another dimension of complexity: existence of *idle*

period. *Idle* period should be effectively incorporated into retrieval schedule in order to minimize the power consumption for multimedia playback.

3 Adaptive Round Merge (ARM) Scheduling

3.1 Multimedia Data Retrieval

Fig. 3 illustrates the multimedia data retrieval operation in desk top environment and mobile environment. The term "desk top environment" is to denote the system which does not have dynamic power management feature. In desk top environment, the disk is active during the entire playback period. In the low power disk drive equipped with dynamic power management feature, it is possible that the disk drive retrieves the data block in burstier manner and goes into the idle state(right hand side of Fig. 3). This is primarily to reduce the fraction of *active period*. We define this policy as *Full Buffering*. Let us examine the Full

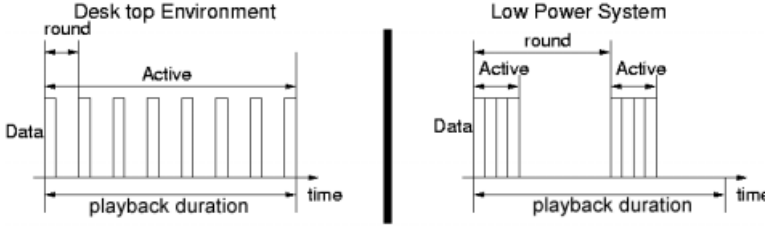


Fig. 3. Multimedia Data Retrieval: Desk top environment vs. Mobile Environment

Buffering strategy in detail. The entire playback is made up of a sequence of *rounds*. In each round, the disk reads the data block at the maximum rate as much as it can and goes into the idle phase. Active period consists of *startup*, *read* and *finish* phases. Let B^* be the size of buffer. The read phase finishes when the buffer is full. The length of read phase in each round, T_p, corresponds to $\frac{B^*}{R-r}$. R and r denotes the maximum transfer rate of the disk and the playback rate. Fig. 4 illustrates the power consumption profile in retrieving the multimedia data using *Full Buffering* strategy. T_s, T_p, T_f and T_0 denotes the length of *startup*, *read*, *finish* and *idle* phase, respectively. T and L denotes the length of a round and the playback duration, respectively. When the playback rate is r, the length of a round, T, can be as large as $\frac{T_p^* R}{r}$. Let us formulate the total power consumption in Full Buffering. Playback consists of a sequence of rounds. The *last* round in the playback may be shorter while the preceding rounds have the same length. Let N be the number of rounds of the same length. We can compute the total power consumption in Full buffering as in Eq. 1.

$$\mathcal{P} = N(P_s + P_f + \alpha T_p) + (P_s + P_f + \alpha T_p^{'})I \tag{1}$$

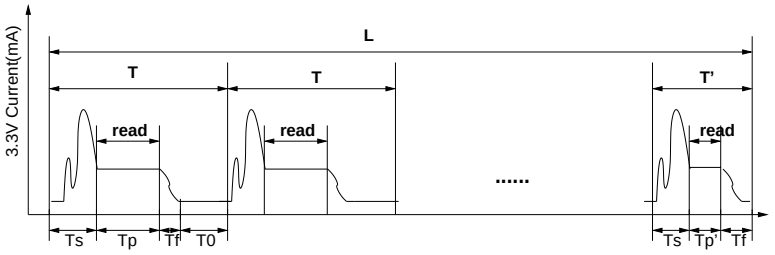


Fig. 4. Multimedia Data Retrieval and Power Consumption Profile

$$= \left\lfloor \frac{Lr}{\frac{B^*}{R-r}} \right\rfloor (P_s + P_f + \alpha \frac{B^*}{R-r}) + (P_s + P_f + \alpha T_p^{'})I \tag{2}$$

P_s and P_f denote the power consumption in *startup* and *finish* phases, respectively. The term α denotes the power consumption rate in read phase. T' is time to read the remaining data in the last round. I is an index function, which is 0 if $L \ mod \ T = 0$ and 1, otherwise. If playback length, L, is an integer multiple of T, i.e. $I = 0$, *full buffering* yields minimum power consumption. $T_p^{'}$ is the length of read phase in the last round.

3.2 Adaptive Round Merge

In *Full Buffering*, the disk drive stops reading by shutdown the disk and goes into idle phase. This is because efficiency of the power consumption is significantly degraded when the disk continues reading after the buffer is full. If the amount of remaining data is *small*, it may not be worth another round which accompanies *startup* and *finish* phase from the perspective of the power consumption. When the buffer is full, we need to decide whether to keep reading the remaining data block in the current round or to enter *finish* phase. We develop a framework which determines to read the remaining data blocks whether in the current round or in the separate round. We assume that the power consumption during the *idle* phase is negligible. Note that T_p is the length of the read phase. The length of the *idle* period, T_0, can be computed as $\frac{T_p * R}{r} - (T_s + T_f + T_p)$. It takes $\frac{B^*}{R-r}$ to fill the empty buffer. Once the buffer is full, disk can retrieve the data block at the rate of consumption. Thus, the amount of data blocks read during T_p can be represented as in Eq. 3.

$$B = \begin{cases} T_p R & , \ if \ \ T_p \leq \frac{B^*}{R-r} \\ \frac{B^*}{R-r} R + (T_p - \frac{B^*}{R-r}) r, & otherwise. \end{cases} \tag{3}$$

Let B the amount of data blocks retrieved during a single round. Then, the amount of remaining data blocks read in the last round, B_l is corresponded to $L \cdot r - NB$. Power consumption in the last round corresponds to $P_s + P_f + \alpha \frac{B_l}{R}$. If we merge the last round with its immediately preceding one(round merge),

we can save the power consumed in *startup* and *finish* phases of the last round. However, *read* phase in the preceding round is extended by $\frac{B_l}{r}$ and additional power consumption, $\frac{\alpha B_l}{r}$ is ensued. Henceforth, we can finally establish function $\mathcal{P}^*$ to determine whether to merge the last round to its immediately preceding round or not as in Eq. 4. If $\mathcal{P}^* > 0$, the last round is merged with the preceding one. Otherwise, last B_l data is retrieved in the separate round. This adaptive round merge algorithm achieves the minimum power consumption in retrieving given multimedia data.

$$\mathcal{P}^* = \underbrace{P_s + P_f + \alpha\frac{B_l}{R}}_{\text{w/o ARM}} \quad \underbrace{- \alpha\frac{B_l}{r}}_{\text{w/ ARM}} \tag{4}$$

The Adaptive Round Merge(ARM) scheduling algorithm does not impose any restriction on whether the medium is detachable [1] or not [2].

4 Simulation Based Experiment

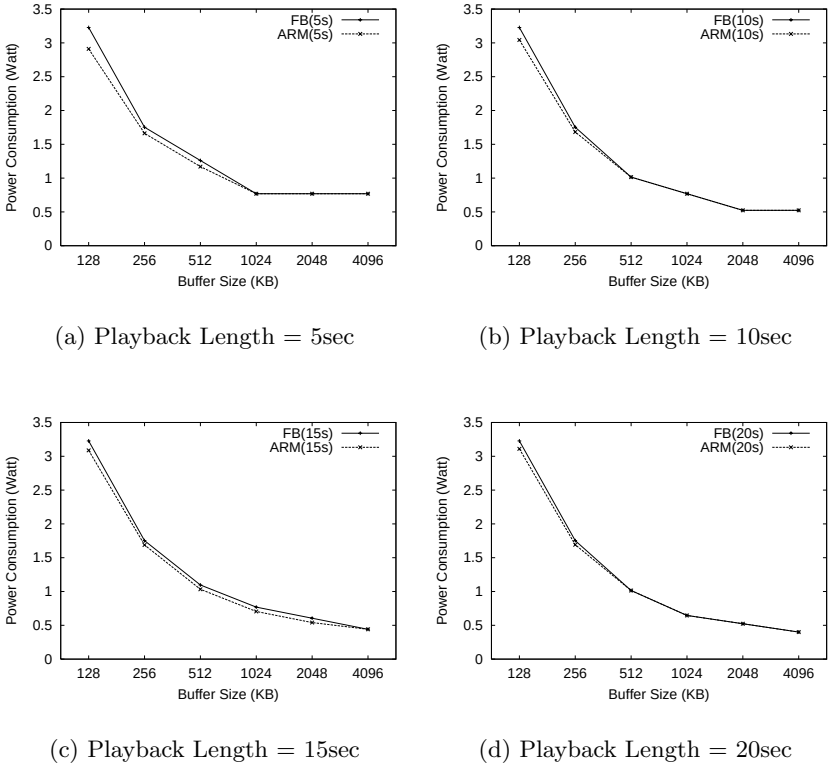
In this section, we compare the power consumption in *Full Buffering* strategy and *ARM*. Disk is modeled after micro-optical disk $DPMO - 501B$ from DataPlay. Simulation parameters are shown in Table 1.

Table 1. Simulation Parameters

Phase	Time	Power Consumption
Startup Phase	1 sec	1.42 Watt
Finish Phase	1 sec	1.04 Watt
α	1.58 Watt/sec	

4.1 Effect of Buffer Size

We first compare the power consumption under different buffer sizes, B^*. The X-axis and the Y-axis in Fig. 5 denotes the buffer size and the power consumption, respectively. The playback rate is 1.4 MBits/sec. As shown in Fig. 5(a), Fig. 5(b), Fig. 5(c) and Fig. 5(d), the power consumption is inversely proportional to the buffer size. The reason is that if the buffer size increases, then the number of rounds decreases that causes the power consumption decreases. When the playback length is 5 sec, the total data size is 896 KByte ($5sec \times 1.4MBits/sec \times 1024 \div 8$). Therefore, it is found that the buffer size beyond 1024 KByte, power consumption is all same among 1024, 2048 and 4096 KByte. In case of the playback length is 10 sec, we could know the identical power consumption for the buffer size of 2048 and 4096 KByte. In Fig. 5(a), the Adaptive Round Merge algorithm brings the reduction of power consumption against Full Buffereing by 10%, 6% and 8% for the buffer size of 128, 256 and 512 KByte, respectively. When the playback length is 20 sec, the power consumption decreases by 4% for



(a) Playback Length = 5sec

(b) Playback Length = 10sec

(c) Playback Length = 15sec

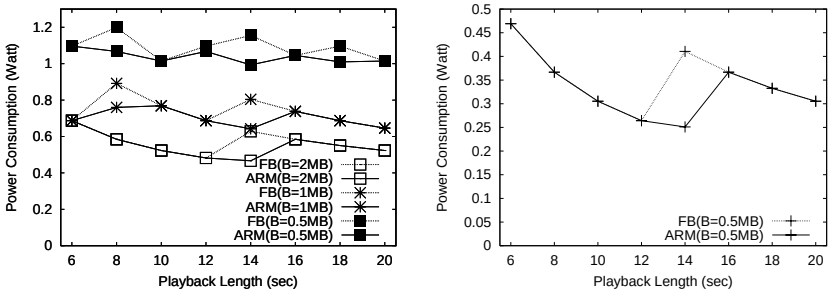
(d) Playback Length = 20sec

Fig. 5. Buffer Size vs. Power Consumption

both buffer size of 128 and 256 KByte. We can observe that allocating sufficient amount of buffer plays a critical role in reducing the power consumption for multimedia playback.

4.2 Effect of Playback Length

Fig. 6 illustrates the relationship between the playback length and the power consumption with the playback rate of 1.4 Mbits/sec and 300 Kbits/sec. The X and Y axis denotes the playback length and the power consumption. In Fig. 6(a), we consider three different buffer sizes, 0.5, 1 and 2 MByte with the playback rate of 1.4 MBits/sec. In Fig. 6(b), we contemplate 0.5 MByte buffer size with the playback rate of 300 MBits/sec. In *Adaptive Round Merge* strategy, it computes the trade off in round merge as in Eq. 4, and merges the last round with the preceding one based on $\mathcal{P}^*$. Therefore, the main savings come from avoiding overhead of spinning up the disk drive in the last round. In Fig. 6(a), when the playback length is 14 sec, ARM can save the power consumption by upto 23% against Full Buffering. When the playback rate is 300KByte/sec, which



(a) Playback rate = 1.4 Mbits/sec (b) Playback rate = 300 Kbits/sec

Fig. 6. Playback Length vs. Power Consumption

is lower than the previous one, the situation is slightly different. A single file where the playback length is 12 sec, has a smaller file size than 500 KByte and thus it is possible to load the entire file into buffer in a single round. In this case, ARM does not bring any improvement against Full Buffering. When the playback length is longer than 12 sec, ARM reduces the power consumption by as much as 40% with 0.5 MByte buffer.

5 Conclusion

In this paper, we present the *ARM(Adaptive Round Merge)* scheduling algorithm to minimize the power consumption for realtime multimedia playback in low power disk drive. In modern low power disk drive, the disk stops spinning the platter and changes to the idle state when it is not in use. Legacy power management strategy in disk drive determines the state of the workload(active or idle) based on *heuristics* or *stochastic prediction*. When the device is designed to perform specific task, e.g. MP-3 player, Personal Video Recorder, Digital Camera, or etc., it is possible to more precisely predict the behavior of I/O workload and can achieve greater savings in power consumption by effectively incorporating it into disk scheduling.

In this work, we focus on retrieving the data block for multimedia playback in more power efficient manner. We investigate the power consumption behavior of low power disk drive and elaborately incorporate the findings in determining the data retrieval schedule for the multimedia data playback. The *ARM(Adaptive Round Merge)* algorithm adaptively minimizes the round to minimize the total power consumption. We find that the ARM algorithm can reduce the power consumption by as much as 23% against legacy Full Buffering algorithm. The Adaptive Round Merge strategy manifests itself when the playback length is relatively short, typically less than 30 sec.

Acknowledgement. Authors would like to thank Dr. Sookyung Kim at LG Electronics for his valuable comments and helpful suggestion on this work.

References

1. http://www.dataplay.com.
2. Ibm family of microdrives. Technical report, IBM Co. Storage Technology Division, 2000.
3. L. Benini, A. Bogliolo, S. Cavallucci, and B. Ricco. Monitoring system activity for os-directed dynamic power management. In *Proceeddings of IEEE International Symposium on Low Power Electronics and Design*, pages 185–190, 1998.
4. E.-Y. Chung, L. Benini, A. Bogliolo, and G. D. Micheli. Dynamic power management for non-stationary service requests. In *Proceedings of Design Automation and Test in Europe*, 1999.
5. F. Douglis, P. Krishnan, and B. Bershad. Adaptive disk spin-down policies for mobile computers. *Computing Systems*, 8:381–413, 1995.
6. P.M. Greenawalt. Modeling power management for hard disks. In *Proceedings of the Second International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems(MASCOTS '94)*, pages 62–66, 1994.
7. D.R. Kenchammana-Hosekote and J. Srivastava. Scheduling Continuous Media on a Video-On-Demand Server. In *Proc. of International Conference on Multi-media Computing and Systems*, Boston, MA, May 1994. IEEE.
8. Yung-Hsiang Lu, Luca Benini, and Giovanni De Micheli. Operating-system directed power reduction. In *Proceedings of the International Symposium on Low Power Design, 2000*. ACM, New York, NY, USA, 2000.
9. Yung-Hsiang Lu and Giovanni De Micheli. Adaptive hard disk power management on personal computers. In *Proceedings of the IEEE Great Lakes Symposium on VLSI, 1999*. IEEE, Los Alamitos, CA, USA, 1999.
10. Dinesh Ramanathan, Sandra Irani, and Rajesh K. Gupta. An analysis of system level power management algorithms and their effects on latency. *IEEE Trans. on Computer-Aided Design of Integrated Cirtuits and Systems*, 21(3):291–305, March 2002.
11. A. L. N. Reddy and J. Wyllie. Disk Scheduling in a Multimedia I/O system. In *Proc. ACM Multimedia Conf.*, pages 225–233. ACM Press, New York, 1992.
12. T. Simunic, L. Benini, P. Glynn, and G. De Micheli. Dynamic power management of laptop hard disk. In *Proceedings of Design, Automation and Test in Europe Conference and Exhibition 2000*, page 736, 2000.
13. T. Simunic, L. Benini, P. Glynn, and G. De Micheli. Event driven power management. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 20(7):840–857, July 2001.
14. IBM Storage Systems Division. Adaptive power management for mobile hard drives. Technical report, IBM Co. Storage Systems Division, San Jose, CA, USA, 1999.
15. Youjip Won and Jaideep Srivastava. "smdp: Minimizing buffer requirements for continuous media servers". *ACM/Springer Multimedia Systems Journal*, 8(2):pp. 105–117, 2000.