Low Power Management of OLTP Applications Considering Disk Drive Power Saving Function

Norifumi Nishikawa, Miyuki Nakano, and Masaru Kitsuregawa

Institute of Industrial Science, the University of Tokyo,
4-6-1 Komaba Meguro-ku, Tokyo 153-8505, Japan
{norifumi,miyuki,kitsure}@tkl.iis.u-tokyo.ac.jp
http://www.tkl.iis.u-tokyo.ac.jp/top/

Abstract. A power-saving management for OLTP applications has become an important task for user budgets and datacenter operations. This paper presents a novel power-saving method for multiple disk drives based on knowledge of OLTP application behaviors. We report detail analysis of power consumption of disk drives and I/O characteristics of OLTP application. We then show experimental and simulation results of our power-saving methods. Our method provides substantially lower power consumption of disk drives compared to that of a conventional OLTP environment.

Key words: Database, Online Transaction Processing (OLTP), Disk Drive, Power-Saving

1 Introduction

Server and storage aggregation at datacenters has increased datacenters' power consumption. The power consumption of servers and datacenters in the United States is expected to double during 2006-2011 [1]. Storage is a high power consuming unit at large datacenters from a database-application workload perspective. Consequently, disk storage power-savings have become a major problem for database systems at datacenters [2, 3].

A Database Management System (DBMS) is reportedly a major storage application [4]. A storage capacity shipment for DBMS is more than 60% of the total shipment of high-end class storage installations, and shipments for online transaction processing (OLTP) such as ERP and CRM constitute more than half of the shipments of storage installations for DBMS. Therefore, storage for OLTP is expected to be major power consumption unit at datacenters. Reducing power consumption of storage devices for OLTP is an important task that must be undertaken to decrease power consumption of datacenters.

Regarding power consumption of a storage unit such as RAID, power consumption of disk drives occupies about 70% of the total storage power consumption [5]. Today's disk drives have a power-saving function such as stopping the spindle motor or parking a head. These functions are useful for power-saving of disk drives; however, several hundred joules of energy and more than 10 s are

required to spin up the disk drives [6]. An inappropriate usage of the power-saving function therefore the increment of power consumption of disk drives and slows down applications. Consequently, it is important to select appropriate opportunities for using power-saving functions to reduce the disk-drive power consumption.

In the past few years, several studies have addressed these problems. The features of these studies are estimation of I/O-issued timing by analyzing application behavior. The length of the estimated period of time is limited by the length (latency) of the transaction. These approaches are, therefore, suitable for long-term transactions and not for short-term transactions. Consequently, it is difficult to apply these approaches to OLTP applications for which the transaction latency time is less than a few seconds.

Workloads of applications at a large datacenters are mainly short-term transactions such as banking or stock-market applications: OLTP applications. Power-saving methods using characteristics of I/O behavior of OLTP, however, have never been examined. In this paper, we focus on the most challenging problem of power saving of disk drives under the unfavorable OLTP environment that all disk drives are likely to be accessed constantly and equally.

Our contribution is to propose of a new power-saving method without OLTP performance degradation by considering OLPT I/O behaviors. The feature of our approach is to extend idle periods of disk drives using the comprehensive behavior of OLTP DBMS. Our approach uses I/O behavior knowledge of OLTP applications and background processes of DBMS. The other contribution is that we measured actual power consumption of OLTP applications on multiple disk drives using a power meter in detail. Few reports describe actual measurements of power consumption.

Our power-saving method enables reduction of disk drive power consumption of more than 38% in the best case in our experimental results. We also intend to use our proposed method to apply large RAID storage systems in future works.

2 Related Works

In this section, we describe related works of storage power-saving methods. These methods are classified into disk drive rotation control methods, I/O interval control methods, data placement control methods.

Disk drive control method controls a disk drive rotation speed or power status. Proposed approaches of disk drive control are categorized into two groups: i) changing the length of wait time to change the status of disk drive to standby or sleep [7, 8]; ii) rotating a disk drive at multiple speeds [9, 10]. These approaches typically use a long period of idle time such as 30 s. Moreover, application-aware power saving methods are also proposed to control disk drives precisely by using knowledge of applications [18]. However, the OLTP application executes multiple transactions in a few seconds and its idle lengths of I/O becomes less than one second, so it is difficult to apply these approaches independently.

I/O interval control method controls the I/O timing of an application to increase the chance that a disk drive is in a power-saving mode. A feature of this approach is to increase the idle period using hierarchical memory architecture such as cache memory [11–13], and changing the application codes in order to control I/O timing [19, 20]. These approaches are useful for applications with a small data footprint size or low I/O frequency, or a long-term transaction. Therefore, it is difficult to apply these methods to OLTP directly.

Data placement control method is intended to reduce the power consumption of disk drives by controlling data placement on disk drives. The idea of this approach is to concentrate frequently accessed data into a few disk drives, then to move the status of other disk drives to standby or sleep [14–17]. These approaches are also useful with applications for which the I/O frequency is low. It is not easy to apply these approaches to OLTP applications since there are many short transactions issued within few seconds. Consequently, it is necessary to combine other approaches that find less frequently accessed data at block level.

3 Characteristics of Disk Drive Power Consumption

This section explains characteristics of power consumption of disk drives based on actual measurement results.

3.1 Measurement Environment

A load-generating PC provides power to a measured disk drive using 4-pin power cables. We connected a digital power meter (WT1600; Yokogawa Electric Corp.) to the power cables in order to measure the electric current and voltages of the disk drive. The load-generating PC CPUs are two Athlon 64 FX-74 3 GHz, 1MB cache, 4-core processors (Advanced Micro Devices, Inc.). Main memory sizes of the load generating PCs are 8 GB. Measured disk drive is Barracuda ES ST3750640NS (750 GB, 7200 rpm; Seagate Technology LLC). The disk drive write caches are turned off to protect reliability of the database because DBMS uses no write cache.

3.2 Power Consumption at Active/Idle States

Fig. 1 depicts a relation between the disk drive power consumption and I/Os per second (IOPS). The I/O size is 16 KB, which indicates that the power consumption of random I/O increases in accordance with an increase of IOPS, but saturates the increase of power where IOPS is larger than 70-80 IOPS. The power consumption of a sequential write is much less than that of random I/O because the disk drive head movements are far fewer than those of random I/O.

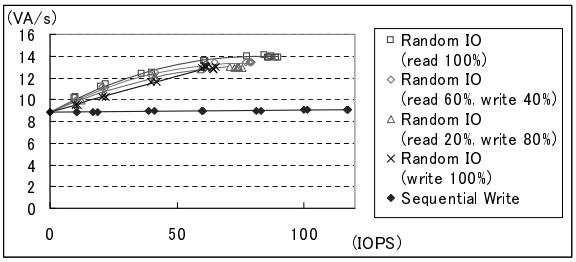

Fig. 1. Relationship between the disk drive power consumption and I/Os per second (IOPS). The power consumption of random I/O increases in accordance with an increase of IOPS. The power consumption of a sequential write is much less than that of random I/O.

3.3 Power Consumption of Standby Status and Break Even Time

We also measured the power consumption of a standby state disk drive, along with the transition from active/idle status to standby status, migration from standby status to active/idle status, and break-even time of the disk drive.

The power consumption of a standby state was 1.5 VA. The transition status from a standby to an active/idle, however, requires 8 s and more than an average of 23.3 VA. The transition status from an active/idle to a standby requires 3.5 VA with 0.2 s. The break-even time calculated from these values is 15.8 s. Therefore, idle time of more than 24 s (15.8 s + 8.0 s) is necessary to use this disk drive power-saving function. Hereafter, we call this idle time as "required idle time".

4 I/O Behavior of OLTP Application

For investigating a power-saving method using characteristics of I/O behavior of OLTP application, we measured I/O behavior of tpcc-mysql [22] on our test bed environment. Here, tpcc-mysql is a simple implementation of the TPC-C benchmark.

4.1 Experimental Environment

The hardware is the same configuration described in 3.1. The software configuration is the following: the OS is 32-bit version of CentOS; the DBMS is MySQL Communication Server 5.1.40 for Linux; and the OLTP application is tpcc-mysql. The file system cache and the disk drive are disabled. The size of the DBMS buffer is 2 GBytes. Our first target is high transaction throughput OLTP applications served at large datacenters. Therefore, we configured the size of DBMS buffer as larger than the size of the database.

The database is approximately 1 GByte (number of Warehouse is 10), in which the Log data size is not included. We partition a disk into 10 volumes and

format these volumes using the Ext2 file system. Log data and each file of the tables and indexes are placed separately into each volume. This file placement eases the receipt of I/O performance data of each database data: we simply measure the OS level performance.

4.2 Behavior Characteristics of TPC-C

Fig. 2(a) shows the number of reads and writes per second and the average I/O intervals of each datum. As presented there, the characteristics of TPC-C I/Os were the following: i) write I/O to Log data were dominant, ii) I/Os to tables and indexes were fewer than two I/Os per second, and iii) more than half these I/Os were writes. No I/O was measured to District, Item, or Warehouse data. The I/O intervals of these data (Log, tables and indexes) were shorter than the required idle length (24 s) except for NewOrder data.

Fig. 2(b) portrays the intervals of I/Os of data. This figure reveals that the idle lengths of OrderLine, Orders, and Stock data were skewed; some idle period was longer than the required idle time. Therefore, a power-saving method using I/O behavior characteristics enables stoppage of disk drives for a long time.

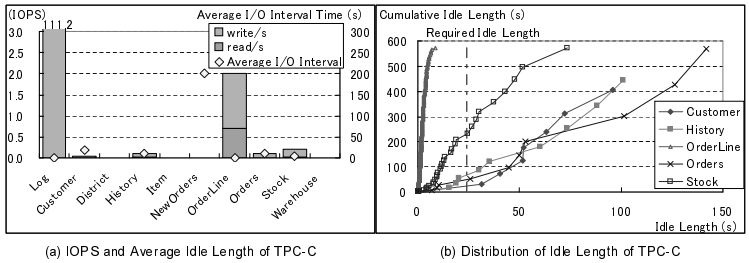

Fig. 2. IOPS, average idle length, and distribution of idle length of TPC-C. The average I/O intervals of data is shorter than the required idle length, some idle period was, however, longer than the required idle time.

5 Power-Saving Method using I/O Behavior Characteristics of OLTP Application

We propose a power-saving method using characteristics of the I/O behaviors of TPC-C application. The features of the proposed method are: i) to generate non-busy disk drives by gathering data of a few I/Os, ii) to delay writing I/Os to database data until the database data are read on the same disk drives.

5.1 Power Saving Method using Data Placement

This method gathers frequently accessed data into a few disk drives, and generate chance to spin down other disk drives which store infrequently accessed data.

As shown in Fig. 2(b), we can observe long I/O intervals for Orders, History, Customer, and Stock data. We expect that this long I/O interval will enable us to use of the power-saving function of disk drives aggressively. This prediction of long I/O intervals cannot be achieved solely considering the I/O disk drive frequency of storage-level knowledge.

5.2 Power-Saving Method using Delayed Write I/Os

We propose a delayed write I/O method to use long I/O intervals for power-saving of disk drives. This method is based on the DBMS behavior of writing operations.

Fig. 3 presents our proposed method. The main idea of our approach is to produce a long idle period by delaying deferred write I/O of database data until the DBMS reads database data on the same disk drive or a checkpoint. As shown there, our approach causes no delay of query processing threads. We can spin down the disk drive at this long idle period without degradation of OLTP throughput. The delay period should be defined based on the dirty page rate of DBMS buffer, the number of pages updated per second, and the interval length of a DBMS checkpoint. This subject warrants future study.

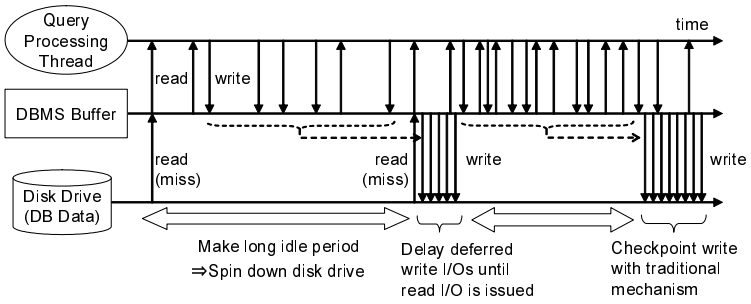

Fig. 3. Mechanism of delayed write I/O. Write I/Os to a disk drive are delayed until the DBMS reads database data on the same disk or a checkpoint.

6 Evaluation

6.1 Evaluation of Data Placement Method based on Access Frequency

For evaluation of two disk drives, we add a SATA disk drive to the configuration described section 3 and 4 and put database data and Log data into two SATA disk drives. The added disk drive is the same model as those described in sections 3 and 4. We connected a digital power meter to the added drive and measured

the disk drive power consumption. The configurations of DBMS and DB are as described in section 3 and 4. Disk drives are configured to transition to standby status when the idle period is longer than 5 s, and move to active state when an I/O arrives.

Data Placement Variation We evaluated four data placements of two disk drives listed in Table 1. The disk drives are of two types: active and inactive. In this approach, disk drive #1 is active and disk drive #2 is inactive.

Table 1. Data Placement (Two Disk Drives)

Case	Data on Disk Drive #1 (Active)	Data on Disk Drive #2 (Inactive)
Case #1	Log	Customer, District, History, Item, NewOrders, OrderLine, Orders, Stock, Warehouse
Case #2	Log, OrderLine	Customer, District, History, Item, NewOrders, Orders, Stock, Warehouse
Case #3	Log, NewOrders, OrderLine, Orders, Stock	Customer, District, History, Item, Warehouse
Case #4	Log, Customer, NewOrders, OrderLine, Orders, Stock	District, History, Item, Warehouse

Evaluation Results of Data Placement Method Fig. 4 shows the actual measured power consumption and transaction throughput of the two disk drives using the data placement, that is, data are distributed into two disk drives in Table 1. Here, we call the results using a spin down function of disk drive as "with power-saving", the results which the spin down function are turned off as "without power-saving".

As shown Fig. 4(a), the power consumption without power-saving is nearly equal to 18 VA for all cases. On the other hand, the power consumption with power-saving method is quite different among four cases. In case #1, the value is equal to the value without power-saving method. This means that in case #1, the data placement method does not reduce the power consumption of disk drives. In case #2, the power consumption is increased to 20.2 VA. In case #3 and #4, on the other hand, the power consumption values of disk drives are smaller than those without power-saving method. In most efficient case, case #4, the power consumption of the disk drives is approximately 23.1% smaller than those without the power-saving method. In Fig. 4(b), the transaction throughput with power-saving of case #1 is 196.3 transactions per second, and is nearly equal to the transaction throughput without power-saving. In case #2, the transaction throughput with power-saving is 144.6 transactions per second, and is shown to drop more than 22% compared with case #2 without power-saving. On the

other hand, in cases #3 and #4 with power-saving, reduction of the transaction throughput keeps approximately 10% degradation.

From Fig. 4, we can find that the considerable data placement can achieve large power saving. In case #1, the power-saving function does not reduce the power consumption. This is because the all of idle length of the disk drives #1 and #2 are less than the standby timeout (5 s). In case #2, the power consumption is increased and transaction throughput is decreased because the length of the idle period of disk drive #2 is longer than the standby timeout but shorter than the required idle time (24 s). This causes an energy loss to spin up disk drive #2 and a transactions delay until the disk drive is spin up. Furthermore, this transaction delay stops I/Os to disk drive #1, causing another energy loss. In cases #3 and #4, the disk drive power consumption is reduced when using our proposed method because the idle periods are longer than the total length of required idle time (24 s). The degradation of transaction throughput was caused by waiting for disk drive #2 to spin up.

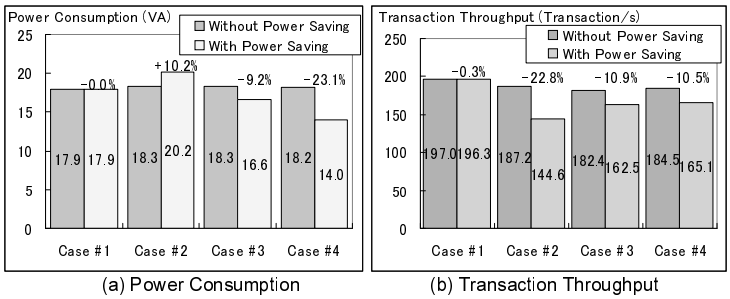

Fig. 4. Measured values of power consumption and transaction throughput (Two Disk Drives for DBMS). Proposed method reduces the disk drive power consumption by 23.1% with only 10% degradation of the transaction throughput.

6.2 Evaluation of Delayed Write I/O

We evaluated our proposed method with a delayed write I/O function. There is no implementation of a delayed write I/O function on commercial DBMS. Therefore, we simulate the I/O behavior of the delayed write I/O function using I/O trace information obtained from the experiments of data placement method. Then, we calculated power consumption and transaction throughput based on these I/O results.

Fig. 5 portrays the results of the disk drive power consumption and transaction throughput for each case. This result is normalized by the power consumption without power-saving. Here, the power-saving method contains both the data placement method and the delayed write I/O method.

As shown in Fig. 5(a), the disk drive power consumption is decreased except in case #1. The maximum reduction of power consumption was 37.9% for cases #3 and #4. Transaction throughput was reduced by 6% in case #1, 5.3% in case #2, and less than 1% in cases #3 and #4(Fig. 5(b)). The power consumption was increased and transaction throughput was decreased in case #1 for the reason described in the preceding subsection.

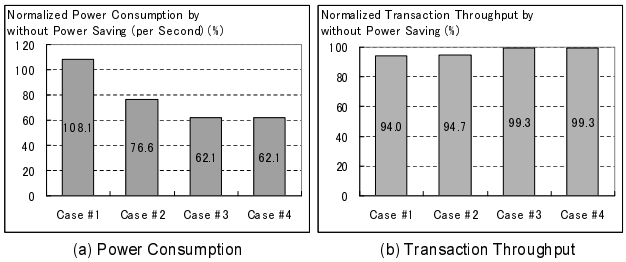

Fig. 5. Power consumption and transaction throughput under two disk drives for DBMS with the delayed write I/O method. The power consumption of disk drives is reduced by 37.9% with little transaction throughput degradation.

7 Conclusion and Future Works

As described in this paper, we measured the actual power consumption values of disk drives and considered the behavior of the TPC-C application in detail. We then proposed a novel power-saving method that enables reduction of power consumption of disk drives for TPC-C applications. The salient feature of our approach is to extend idle periods of disk drives using a data placement method and delayed write I/O method based on a comprehensive behavior of OLTP DBMS executing multiple transactions. We demonstrated that our method achieves an approximately 38% reduction of the disk drive power consumption for a TPC-C application without decreasing its throughput.

References

1. U.S. Environmental Protection Agency ENERGY STAR Program, Report to Congress on Server and Data Center Energy Efficiency Public Law 109-431, `http://www.energystar.gov/ia/partners/prod_development/downloads/EPA_Datacenter_Report_Congress_Final1.pdf`.
2. Bauer,R.: Building the Green Data Center: Towards Best Practices and Technical Considerations. In: Storage Networking World Fall 2008 Conference, `http://net.educause.edu/ir/library/pdf/bauer.pdf` (2008)
3. Chu,P.B. Riedel,E.: Green Storage II: Metrics and Measurement, `http://net.educause.edu/ir/library/pdf/churiedel.pdf` (2008).

4. Reinsel,D: White Paper Datacenter SSDs: Solid Footing for Growth, IDC #210290 (2008)
5. Poess,M. and Nambiar,R.O: Energy cost, the key challenge of today's data centers: a power consumption analysis of TPC-C results. In: International Conference on Very Large Data Base, pp.1229-1240 (2008)
6. Product Manual Barracuda ES Serial ATA, http://www.seagate.com/staticfiles/support/disc/manuals/enterprise/Barracuda\%20ES/SATA/100424667b.pdf (2006)
7. Douglis,F., Krishnan,P., Bershad,B.: Adaptive Disk Spin-Down Policies for Mobile Computers. In: Proceedings of 2nd USENIX Symposium on Mobile and Location Independent Computing (1995)
8. Helmbold,D.P., Long,D.D.E., Sconyers,T.L., Sherrod,B.: Adaptive Disk Spin Down for Mobile Computers. In: Mobile Networks and Applications, Vol. 5, No. 4 (2000)
9. Gurumurthi,S., Sivasubramaniam,A., Kandemir,M., Franke,H.; DRPM: Dynamic Speed Control for Power Management in Server Class Disks. In: 30th Annual International Symposium on Computer Architecture (2003)
10. Zhu,Q., Chen,Z., Tan,L., Zhou,Y., Keeton,K., Wilkes,J.: Hibernator: Helping Disk Arrays Sleep through the Winter. In: Proceedings of Twentieth ACM Symposium on Operating Systems Principles (2005)
11. Papathanasiou,A.E., Scott,M.L.: Energy Efficient Prefetching and Caching. In: Proceedings of the USENIX 2004 Annual Technical Conference (2004)
12. Li,D., Wang,J.: EERAID: Energy Efficient Redundant and Inexpensive Disk Arrays. In: Proceedings of 11th Workshop on ACM SIGOPS European workshop (2004)
13. Yao,X., Wang,J.: RIMAC: A Novel Redundancy based Hierarchical Cache Architecture for Energy Efficient. In: Proceedings of High Performance Storage System 2006 EuroSys Conference (2006)
14. Colarelli,D., Grunwald,D.: Massive Arrays of Idle Disks for Storage Archives. In: Supercomputing, ACM /IEEE 2002 Conference (2002)
15. Pinheiro,E., Bianchini,R.: Energy Conservation Techniques for Disk Array Based Servers. In; Proceedings of 18th Annual International Conference on Supercomputing (2004)
16. Weddle,C., Oldham,M., Qian,J., Wang,A.A.: PARAID: A Gear-Shifting Power-Aware RAID. In: FAST'07: 5th USENIX Conference on File and Storage (2007)
17. Son,S.W., Chen,G., Kandemir,M.: Disk Layout Optimization for Reducing Energy Consumption. In: Proceedings of 19th Annual International Conference on Supercomputing (2005)
18. Gniady,C., Hu,Y.C., Lu,Y.H.: Program Counter Based Techniques for Dynamic Power Management. In: High Performance Computer Architecture (2004)
19. Heath,T., Pinheiro,E., Hom,J., Kremer,U., Bianchini,R.: Application Transformations for Energy and Performance-Aware Device Management. In: Parallel Architectures and Compilation Techniques (2002)
20. Son,S.W., Kandemir,M., Choudhary,A.: Software-Directed Disk Power Management for Scientific Applications. In: Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium (2005)
21. Transaction Processing Performance Council, TPC-C, an online transaction processing benchmark, http://www.tpc.org/tpcc/.
22. tpcc-mysql, https://code.launchpad.net/~perconadev/perconatools/tpcc-mysql.