Placement of VBR Video Data on MZR disks

Jisun Kang and Heon Y. Yeom
Department of Computer Science
Seoul National University
Seoul, 151-742, Korea

Abstract

Recently, disk zoning techniques have been widely used for increasing disk drive capacity. As a side effect, data transfer rates for outer zones of a hard disk are much higher than those of inner zones. Without giving proper care to disk zoning features, disk bandwidth cannot be utilized effectively. In this paper, we propose a data placement scheme for VBR (Variable Bit Rate) video objects, called NCTT (Near Constant Transfer Time). NCTT considers the nature of the VBR, the MZR disk characteristics and the popularity of video objects to effectively place the blocks. In NCTT, the disk blocks of a video object are placed so as to have similar transfer times, i.e, it takes a similar time to retrieve each disk block. Through extensive simulations, we compared the performance of NCTT with existing schemes. The simulation results show that NCTT utilizes the disk bandwidth effectively and supports more users than existing schemes.

1 Introduction

The older hard disks have a CAD (Constant Angular Density) structure so that every track has identical storage capacity and transfer rate. Since the transfer rate is fixed, there is no difference in the performance whether a block is placed in the outer zone or in the inner zone. On the other hand, new hard disks utilize a disk zoning (or Multi-zone Recording (MZR)) technique to increase the disk capacity. The basic idea is to divide a disk surface into zones such that cylinders in the outer zones contain more sectors than cylinders in the inner zones[1]. As a side effect, data transfer rates for the outer zones of a hard disk are much higher than those of inner zones. Hence it takes less time to read a block from the outer zones than from the inner zones. If the data blocks are placed on the disk without considering disk zoning features, disk bandwidth cannot be utilized effectively. For example, if a popular video is placed in the inner zone, the low transfer rate of the inner zone results in low performance of the video server.

Another factor to consider in placing video data on the disk is the VBR nature of the video data. In many cases, VBR video data is divided into a series of variable-sized blocks which have the same playback time[16]. To guarantee deterministic service in VOD systems, resources should be reserved based on the worst case assumptions, i.e, buffers and disk bandwidth must be reserved according to the size of the largest block of the object. Therefore, if the largest block is placed in the inner zone, a low transfer rate of the inner zone increases the time to retrieve the largest block, which results in low performance of the video server. Therefore, to effectively place VBR video blocks on an MZR disk, the MZR disk characteristics, the popularity of video objects and the variable block size due to the bit rate variability of VBR video must be taken into consideration.

In this paper, we propose a data placement scheme for VBR video objects called NCTT (Near Constant Transfer Time), considering the three characteristics mentioned above. In NCTT, the placement order of the video objects is determined by their popularity. In placing one video object, NCTT sorts all the blocks of the video object by size and places the largest block in the outermost zone that has enough space, to exploit its high transfer rate. Then NCTT calculates the STRR (Size-Transfer Rate Ratio) for the largest block, which is the ratio of the size of the block to the transfer rate of the zone that the block is placed in. All the other blocks are placed in their proper zones according to the calculated STRR, i.e., all the blocks of one video object are placed so as to have similar transfer times, i.e. the time to retrieve each block. Hence, the variability of the transfer time is reduced, so that the utilization of the disk is increased. Also, since VBR video objects have variable-sized blocks, there is the possibility for the video blocks of less popular video objects to be placed in the outer zone. Consequently, NCTT supports more users compared with other schemes, because it reduces the maximum transfer time, the major criterion in admission control.

The rest of this paper is organized as follows. Section 2 briefly surveys related work, section 3 defines our system model, and Section 4 explains the admission control scheme. In Section 5, we present the NCTT data placement scheme and the performance evaluations of our scheme is given in section 6. We extend our scheme so that it can be used with RAID in Section 7. Finally, we conclude this paper in Section 8.

2 Related Works

To effectively place VBR video blocks on an MZR disk, the popularity of video objects, the variable-sized blocks of VBR video objects and the MZR disk characteristics must be taken into consideration. There have been several studies considering some of these characteristics. Heltzler et al.[4] proposed a scheme grouping the tracks from each disk zone to form a logical track. Disk I/O bandwidth is made to look like uniform by always retrieving a logical track as an access unit. However, the overall disk throughput is low due to excessive seek operations and, since the number of tracks in different zones is not the same, disk space in zones with a larger number of tracks is wasted. Birk[5] presented a track-pairing scheme which groups two complementary tracks into a track pair, and data is placed in units of track pairs. This approach is less restrictive than the previous approach in that it makes a compromise between uniform bandwidth and disk access time. In [6], Ghandeharizadeh et. al proposed the FIXB and VARB for Constant Bit Rate (CBR) streams. In each disk zone, the block size is fixed for FIXB, while the transfer time for a disk block is maintained as a constant for VARB. In both methods, data blocks in outer zone are read ahead to compensate for slower access in the inner zones. However, both methods suffer from excessive unused storage space, long initial delay, and large buffer requirements. To compensate for this drawback, they proposed MaxT and MinW to reorganize the configuration of disk zoning.

The previous approaches consider only disk zoning features and the variability of data transfer rates, without considering the popularity of video objects. Kim et al.[7] proposed a placement scheme for CBR streams considering the popularity of video objects. Two schemes were proposed. The first one, termed Linear, places the video objects in order from the outermost zone according to their popularity. The second one, termed SH, places popular video objects around the frequently positioned cylinders of the disk head.

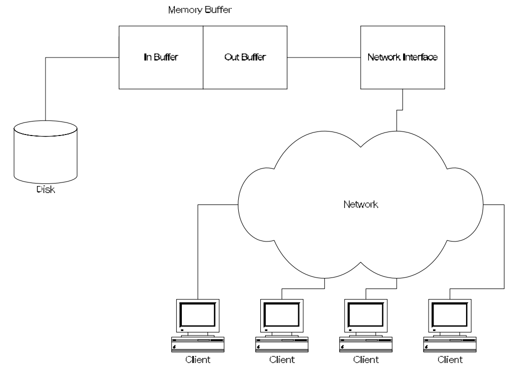

Figure 1: Architecutre of VOD server

In summary, SH places each block in the best position with regard to response time, the time to access each block, to reduce the seek time. However, these schemes are only applicable to CBR video objects.

Wang et al.[8] proposed a data placement scheme to store VBR video on an MZR disk. The scheme employs the concept of CRT (constant read time), that a constant period of time is allocated for retrieving each data block. In other words, the size of each block is proportional to the transfer rate and thus varies in different disk zones. And it determines the number of blocks allocated to each zone in proportion to the capacity of each zone. Finally, it determines the retrieval sequence of the blocks that meets the requirement of jitter-free playback and minimizes the server side buffer. Several algorithms were suggested. However, some algorithms require small computational overhead but result in too large a buffer requirement, and others reduce the buffer requirement but require too much computational overhead. Also, the buffer requirement of the latter algorithm is still much larger than NCTT as will be shown in Section 6.

3 System Model

Figure 1 shows the architecture of a VOD server. A VOD server has three main components: disk, memory buffer, and a network interface. Video objects are stored on the disks, read from the disk to the memory buffer, and sent to the clients via the network. Memory buffer is logically divided into In_buffer and Out_buffer. The In_buffer is used to read data from disks, the Out_buffer to send data over the network.

As we are concentrating on the disk and buffer management, network bandwidth is assumed to be sufficient.

Usually, The Video server reads data from the disk in cycles of a fixed length in time, called a round. A certain amount of data, enough to be played for the round is retrieved from the disk in a round and sent over the network [9][10]. Thus the blocks of the video objects have the same playback time as a round. It is called the CTL (Constant Time Length) method, and it is shown to outperform CDL (Constant data Length) for predominantly read only environment like VOD systems. All the notation in this paper is defined in Table 1.

T	Length of round
T_{rot}	Maximum rotation latency
$r_{i,j}$	Average data rate of jth block of video object i
T^i_{max}	Maximum transfer time of video object i
B_i	Buffer requirement of video object i
Z_j	Data transfer rate of jth zone
C_j	Capacity of jth zone
z	Number of disk zones
C	Total capacity of disk
CYL	Number of total cylinders in disk

Table 1: Parameters and Notations

4 Admission Control in MZR disk

To guarantee the quality of service, buffer and disk bandwidth must be reserved for a new stream before the new stream is serviced[3][16].

4.1 Buffer Constraint

Using double buffering, a disk block is retrieved into In_buffer in a round and that block is sent over the network from Out_buffer in the next round. Therefore, if the reserved buffer size is the maximum sum of two contiguous blocks, there is no buffer overflow in any circumstance. The buffer requirement B_i of a video object i is defined as follows.

$$B_i = T \times \max(r_{i,j} + r_{i,j+1}) \; for\ all\ j \tag{1}$$

A new stream can be accepted if more than B_i buffer space is available.

4.2 Disk Constraint

Provided that the service time to read m blocks from a disk is shorter than a round, we can service m streams without any starvation. Disk service time consists of the seek time, the rotational latency and the data transfer time of the disk [1]. In computing this service time, we have to assume the worst case, so that no matter where the blocks are located on a disk, we will have enough time to read them.

The largest value of the total seek overhead occurs when the segments are equally spaced on the disk, using SCAN disk scheduling [13]. That is, Let m be the number of streams to be serviced and γ be the function that calculates the seek time for a given seek distance. Then the worst case seek time is as follows[1][13].

$$Total\ Seek\ Time = m \times \gamma(CYL/m) \tag{2}$$

Assuming that the jth block of video object i is placed in the kth zone, the worst case transfer time, the maximum transfer time of the video object i is defined as follows.

$$T^i_{max} = T \times \max\{\frac{r_{i,j}}{Z_k}\} \; for\ all\ j,\ k \tag{3}$$

The largest value of the rotational latency can be obtained from the disk specification.

Therefore, if the following equation holds, we can service m streams.

$$Total\ Seek\ Time + \sum_{i=1}^{m}(T^i_{max} + T_{rot}) \leq T \tag{4}$$

5 NCTT placement scheme

It is important to note in Equation 4 that both the total seek time and the rotational latency are constant for a given request. Hence, the performance of the VOD server depends on the aggregate maximum transfer time. Since the maximum transfer time, T^i_{max} usually depends upon the largest block of a video object i, it would be best for the largest block of the popular video object to be placed in the outer zone. However, there is no need for the smaller blocks to be placed in the outer zone since it does no good to read smaller blocks fast. The savings simply can not be used. Due to the VBR characteristic of the video data, the size of the largest block is about three or four times of the average size of the other blocks. That is, we will have enough time to read a small block even though it is placed in the inner zone where the transfer rate is lower.

NCTT utilizes the variability of disk block sizes and the variability in bit rates of the video object to match them accordingly. For a video object, NCTT sorts all the blocks of the video object by size and places the largest block in the outermost zone that has enough space, in order to minimize T^i_{max}. Then, the other blocks are placed in the proper zone so that their transfer times are similar to that of the largest block. To choose an appropriate zone for each block, we calculate the STRR (size to transfer rate ratio) of the largest block and try to make the STRR constant for all the other blocks. Assume that the jth block of the video object i is placed in the kth zone. The STRR of the jth block is defined as follows.

$$STRR = \frac{r_{i,j}}{Z_k} \quad (5)$$

It is impossible to make the STRR of every block constant since the number of blocks in each zone is limited. However, it would be best if we can minimize the variation.

In addition, NCTT places the most popular video object first in order that the large blocks of the more popular video objects exploit the high transfer rate of the outer zone. Thus, disk blocks of one video object are distributed over several zones and disk blocks of less popular video objects have a chance to be placed in the outer zone. That is, T^i_{max} of the less popular video objects is reduced, too. Therefore, NCTT minimizes T^i_{max} for each video object with regard to its popularity, and the variability of the transfer time is reduced, so that the utilization of the disk is increased at the same time.

The placement algorithm for one video object is given in figure 2.

```
NCTT data placement(video object) {
sort blocks of video object by block size;
/* find the outermost zone with enough space */
for each zone i in disk from the outermost
zone {
   if (i has enough space for the largest
    block in the video object) {
      place the largest block in zone i ;
      calculate STRR of the largest block
       for zone i ;
      CRITERIA_VALUE
           = STRR of the largest block ;
      break ;
   }
/* placement of the other blocks */
for each block i in the video object from
the outermost zone {
   for each zone j in disk {
     calculate STRR of block i for zone j;
     if (j has enough space for i &&
       STRR of i ≃ CRITERIA_VALUE )
         place block i in zone j ;
   }
}
```

Figure 2: NCTT placement algorithm

6 Performance Evaluation

We have conducted extensive simulations to analyze the performance of our placement scheme using MPEG-1 traces from [15]. We assumed that the system uses a SCAN disk scheduling algorithm and that an IBM DFMS 5x hard disk is used. The disk characteristics are described in Table 2 .

Capacity	5.3 GB
RPM	5400
Number of zone	13
Maximum Seek Time	16.5 ms
Minimum Seek Time	0.6 ms
Transfer rate of outermost zone	12.58 MB/s
Transfer rate of innermost zone	7.65 MB/s

Table 2: Disk characteristics

Figure 3 shows the performace of each scheme with changes in the Zipf parameter. Assuming that the selection of video objects is characterized by a Zipf

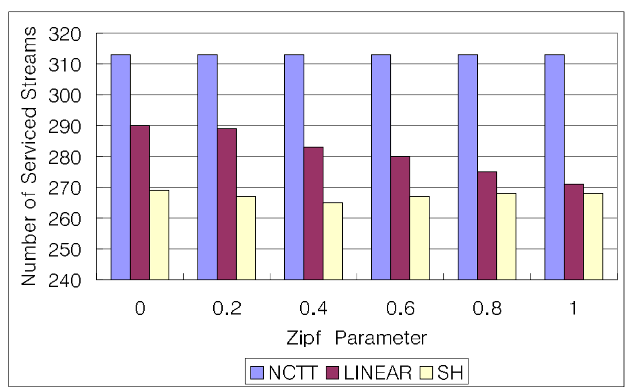

Figure 3: Performance of schemes with different Zipf parameter

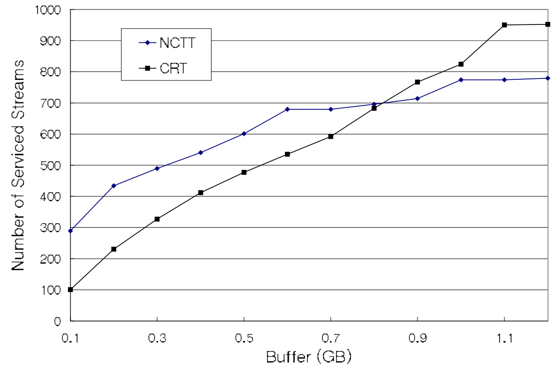

Figure 4: Performance of schemes with different buffer sizes

distribution, the change of Zipf parameter is that of the popularity, access frequency of each video object. As the Zipf parameter increases, the performance of all schemes decreases. However, NCTT shows almost even performance over changes in the Zipf parameter, compared with the other schemes. The result shows that NCTT is more flexible than the other schemes and though the popularity is misestimated, such misestimation won't affect the performance of VOD systems. SH shows the poorest performance, though it has a smaller response time than Linear [7]. The reason is that SH has a larger maximum transfer time than Linear since frequently positioned cylinders of the disk head aren't placed on the outermost zone. Since the performace of CRT does not change with the popularity of video objects, we did not compare the performance of the CRT scheme.

Next, we look at the performance of each scheme with different buffer sizes with CRT in Figure 4. As NCTT, SH and Linear have the same buffer requirement, we only compared NCTT and CRT. NCTT outperforms CRT with small buffer size, while CRT outperforms NCTT with large buffer size. However, CRT requires too much computational overhead. CRT divides a video object into a series of disk blocks that have the same transfer time. That is, the size of each block is exactly proportional to the transfer rate. And the disk blocks are allocated to each zone in proportion to the capacity of each zone. With the number and size of disk blocks per zone, it determines the retrieval sequence of these blocks that meets the requirement of jitter-free playback and minimizes the server side buffer. In [8], three algorithms were suggested to get the retrieval sequence. Though we selected the algorithm that was presented as appropriate in [8], it still requires too much overhead. In the case of some video objects, it took 3 days to calculate the buffer requirement using a Sun Ultrasparc II system.

The average value of buffer requirements for CRT is about two or three times that of NCTT. However, buffer requirement of CRT for some video objects is ten times of the buffer requirement of NCTT. The Buffer requirement and the maximum transfer time of each video object are shown in Table 3. In this experiment, we changed the round length according to buffer size as the optimal round length is changed by buffer size [12] and assumed that the popularity of video objects is uniform.

		buffer requirement (KB)		*maximum transfer time (ms)*	
No.	name of object	NCTT	CRT	NCTT	CRT
1	ASTERIX	415	1858	19.7	10.1
2	ATP	355	842	13.91	9.89
3	BOND	362	711	18.12	10.98
4	DINO	227	517	10.39	5.91
5	FUSS	381	672	20.25	12.25
6	MOVIE	270	966	12.51	6.45
7	MTV	567	1578	23.82	11.11
8	MTV_	561	8461	28.05	8.94
9	NEWS_	497	1182	20.83	6.94
10	RACE	583	904	24.93	13.89
11	SBOWL	373	854	15.54	10.62
12	Talk	221	424	8.79	8.75

Table 3: Comparison between CRT and NCTT

Finally, the storage utilization of all schemes is the same at 95 %. Storage utilization isn't 100 % as we placed one video object in a disk only if the disk has enough space for one object. We could place 12 MPEG-1 video objects on a single disk.

7 Placement on RAID

Our placement scheme can be easilly extended to RAID. we apply a staggered striping scheme [17] when placing the first block of the video object to balance the load among the disks and adopt RTL (Round Time Length) striping to guarantee the quality of service in RAID [12]. In RTL, if k active streams are being serviced from disk i at the current service round, then these streams will be serviced from disk $(i+1)\ mod\ n$ at the next round where n is the number of disks in the disk array. That is, if the first block of one video object is placed on a certain disk, the disks for the other blocks to be placed are determined. Placement method for a single disk is the same method as the scheme described in Section 3. In other words, the blocks are placed on an appropriate zone of the disk, according to the STRR of the largest block. As the experimental results are the same as using only a single disk, the results aren't presented in this paper.

8 Conclusion

Various schemes considering disk zoning features, the popularity of video objects, or VBR characteristics were presented, but there is no scheme considering all three features. To effectively place VBR video blocks on an MZR disk, all of them must be taken into consideration. In this paper, we have proposed a data placement scheme for VBR video objects called NCTT (Near Constant Transfer Time), considering all of them. We described an admission control strategy a MZR disk to guarantee the quality of service, presented an NCTT algorithm using that strategy, and extended the NCTT scheme to be used with RAID. Through extensive simulation experiments, we have shown the performance of our new scheme is superior to the other existing schemes. The simulation results show that NCTT has a good performance with small buffer size and less overhead.

References

[1] Chris Ruemmler and John Wilker. An Introduction to disk drive modeling. IEEE Computer, 27(4), pages 17-28, March 1994.

[2] E.Chang and A. Zakhor. Admission control and data placement for VBR video servers. In Proc. of IEEE Int'l conference on Image Processing, pages 278-282, Austin, June 1994.

[3] J.Dangler, E. Biersack, and C. Bernhart. Deterministic admission control strategies in video servers with variable bit rate. In Proc. of International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services, pages 164-171, Hiroshima, Japan, June 1996.

[4] S.R. Heltzler, J.M. Menon, and M.F. Motoma. Logical data tracks extending among a plurality of zones of physical tracks of one or more disk devices. U.S. Patent No. 5202799, April 1993.

[5] Yitzhak Birk. Track-pairing: a novel data layout for vod servers with multi-zone-recording disks. In Proc. of IEEE Conf. on Multimedia Computing Systems, pages 248-255, 1995.

[6] Shahram Ghandeharizadeh, Seon Ho Kim, Cyrus Shahabi, and Roger Zimmermann. Placement of continuous media in multi-zone disks. Multimedia Information Storage and Management, 1996.

[7] Jeong-Won Kim, Young-Uhg Lho, and Ki-Dong Chung. An Effective Video Block Placement Scheme on VOD Server Based on Multi-zone Recording Disks. In Proc. of IEEE Conf. on Multimedia Computing Systems, pages 29-36, June 1997.

[8] Yu-Chung Wang, Shiao-Li Tsao, and Ray-Yi Chang. A Fast Data Placement Scheme for Video Servers with Zonned-Disks. In Multimedia Computing and Network, pages 92-102, 1997.

[9] A.L. Narasimha Reddy and James C. Wyllie. I/O issues in a multimedia system. IEEE Computer, 27(3), pages 67-74, March 1994.

[10] D.J. Gemmel, H.M. Vin, D.D. Kandlur, P.V. Rangan, and L.A. Rowe. Multimedia storage servers: A tutorial and survey. IEEE Computer, 28(5), pages 40-49, May 1995.

[11] Ed Chang and Avideh Zahkor. Cost analyses for VBR video servers. In Proc. of SPIE Multimedia Computing and Networking, pages 381-397, 1996.

[12] KyongOh Lee and Heon Y. Yeom. Deciding Round Length and Striping Unit Size for Multimedia Servers. In Proc. of the 4th International Workshop on Multimedia Information Systems (MIS'98), pages 33-44, September, 1998.

[13] Edward Chang and Hector Garcia-Molina. Effective Memory Use in a Media Server. In Proc. of VLDB'97, pages 496-505, August 1997.

[14] D. Kotz, S.B. Toh, and S. Radhakrishnan. A detailed simulation model of the hp 97560 disk drive. Dartmouth College Technical Report PCS-TR94-220, 1994.

[15] ftp://ftp-info3.informatik.uni-wuerzburg.de /pub/MPEG

[16] KyongOh Lee and Heon Y. Yeom. An Effective Admission Control Mechanism for Variable-Bit-Rate Video Streaming. to appear in ACM Multimedia Systems.

[17] S. Berson, S. Ghandeharizadeh, R. Muntz, and X. Ju. Staggered striping in multimedia information systems. In Proc. of ACM SIGMOD, pages 79-60, Minnesota, USA, June 1994.