CMPSCI 754 Multimedia Systems **Spring 2001**

Lecture 6: 20 Feb

Lecturer: Prashant Shenoy *Scribe: Rahul Gupta*

Placement Techniques

The advantages of striping the multimedia files across multiple disks in disk array are concurrency and load balancing. Two important issues crop up while striping the multimedia file.

- What should be the striping unit size?
- What should be the degree of striping? (i.e, how many disks will contain the stripped multimedia file?)

In this lecture, we will look at various factors which affect the optimal block/striping unit size and then build an analytical model for finding optimal block size.

Factors affecting Optimal Block Size

The factors that affect the optimal block size are

1. Server Configuration
   - Number of disks in the array
   - Physical characteristics of disks
   - Type of disk array (i.e, redundant vs non-redundant array)
2. Client Characteristics
   - Number of clients accessing the server
   - Client request size

We will determine the optimal block size for given server configuration and client characteristics (the optimal block size varies as server configuration and client characteristics varies). We have to select a metric to determine optimal block size. Before going on, lets see what is the metric for determining optimal block size for text files. The characteristics of text file requests are :

- The accesses are aperiodic (sequential or random). This asks for client-pull architecture.
- The accesses are best effort. They don't have any real time deadlines. So, the metric here is minimizing response time.

Now for continuous media case, the various characteristics are :

- The accesses are periodic and sequential accesses. So, the server-push architecture is appropriate.

- The accesses are real time. They have timing deadlines and any access of a block after deadline is useless for the client. So, the metric here is minimizing the tail of response time distribution which maximizes the number of clients satisfied.

The metric that we choose for determining optimal block size is to minimize the service time of the most heavily loaded disk which is the bottleneck for number of clients satisfied. If we change the metric, the optimal block size changes.

Fixed size blocks

We can have fixed as well as variable block sizes. For today's lecture, we will consider fixed size blocks for finding optimal block size. First, let us see the affect of stripe unit size on load imbalance and latency overhead.

1. If the stripe unit size is small, then we have lot of accesses for a particular client in any given round so this increases the latency overhead. Also, now since the blocks for this multimedia stream are more spread across disks, the load imbalancing is low.
2. If the stripe unit size is large, then we have less number of accesses for a particular client in any given round so this decreases the latency overhead. Also, now since the blocks for this multimedia stream are less spread across disks, the load imbalancing is high.

So, there is tradeoff between load imbalance and latency overhead when stripe unit size changes. So, we have to find optimal block size that balances these tradeoffs.

There are three approaches we can take to do this. They are simulation, analysis and by actual experiment. We will take the analytical approach to develop an analytical model to compute the service of the most heavily loaded disk.

Analytical Model

- Given : Server configuration and client characteristics.
- Objective : Predict the service time of the most heavily loaded disk.
- Main Steps :
  1. Estimate the number of blocks accessed from a disk by a single client
  2. Estimate the total number of blocks accessed from a disk.
  3. Estimate the number of blocks accessed from the most heavily loaded disk.
  4. Compute the service time of the most heavily loaded disk.

With the help of an example, let us construct a very simple analytical model as follows.

1. Let C_i be the i^{th} client. Frame size distribution is used to determine the total number of blocks accessed by C_i from the disk array. Let $P(C_i = k) = {b_i}^k$.
2. Let ${X_i}^j$ be the number of blocks accessed from disk j by client i. Let us calculate the probability of ${X_i}^j = k$. We can assume that a request is equally likely to start on any of the D disks.

   $P({X_i}^j = 1) = \frac{{b_i}^1}{D} + \frac{2.{b_i}^2}{D} + ... + \frac{D.{b_i}^D}{D}$

Note that, this equation stops after D^{th} term but ideally it should go to $2D-1$ term but since we are considering very simplified model, we neglect the last $D-1$ terms.

$P({X_i}^j = k) = \sum_{m=1}^{D} {b_i}^{(k-1)D+m} . \frac{m}{D} (k = 1, 2, 3....)$

3. Total number of blocks accessed from disk j by n clients is $Y^j = \sum_{i=1}^{n} {X_i}^j$

4. Number of blocks accessed from the most heavily loaded disk is $Y^{max} = max(Y^1, Y^2, ..., Y^D)$

5. Service time of the most heavily loaded disk $\tau = E(Y^{max}) * (t_s + t_r + t_x)$ where

   t_s : seek time to access a block
   t_r : rotational latency to access a block
   t_x : transfer time of a block

Validation of the model

Note: the various plots are given in slides for this lecture.
For plots in slide 60:

1. The optimal block size is the one that corresponds to the minima of the service time graph. In the plots, the optimal size is around 80kB (service time is 350msec).

2. We can also define $x\%$ optimal envelope to be the block sizes which yield service times within $x\%$ of the optimal. 5% optimal envelope corresponds to 64kB-150kB.

Effect of Quality of Service Requirements

For plot in slide 62.

1. Optimal block size decreases with increase in QoS requirements (that is, percentile of service time in plot).

2. So, for a stringent QoS, ad-hoc techniques that choose a large stripe unit size can reduce the number of clients supported.

Effect of the number of the clients

For plots in slide 63

1. As the number of clients increases, optimal block size increases as increase in block size leads to more load imbalance which is offset by increase in number of clients.

2. Small block size yields uniform distribution of load for small number of clients.

3. Large block size yields higher disk throughput for large number of clients.

Effect of Data rate

For plot in slide 64.

1. As we increase the data rate, optimal block size increases. This is also intuitive because as we increase the data rate, there will be more and more accesses in any given round and the load imbalance caused by increasing optimal block size is offset by increase in number of accesses.

Effect of number of disks

For plot in slide 65

1. As we increase the number of disks, the optimal block size decreases. This is also intuitive because the decrease in optimal block size leads to more number of blocks that are more spread out among increased number of disks to provide more load balancing.

Selecting a block size

The optimal block size depends on design goal, server configuration and client characteristics. For example, suppose we want to minimize service time over a range of workloads.

1. Client characteristics such as data rate may be unknown at design time. So, the block size must provide good performace over a range of such parameters.
2. Determine the $x\%$ envelope for different workloads.
3. Choose a block size that is contained within the optimal envelope for all workloads.

As an example, clients can vary from 5 to 100.
data rate can vary from 10 frames/sec to 30 frames/sec.
bit rate can vary from 300kb/s - 4Mb/s

For each case, find the range of block sizes with in $x\%$ (x=5) optimal envelope and take the intersection of these ranges. Then, we can choose the block size from the final intersected range as this provides good performance within 5% of optimal envelope for various workloads.