CSE102 : File and Storage Structures

Solution Set for Assignment 2

1. In this question you have to find the service times for small as well as large reads, writes and read-modify-writes for a 6 disk RAID level 5 system. The minimum and maximum seek times for a disk are 4ms and 16ms respectively. The disk rotates as 5400 rpm and each track contains 65 KB divided into 512 byte sectors. These metrics for a disk drive have been chosen to be same as that for a disk available in the market.

   Now, as we know for a disk,

   Access time = Seek time + Rotational Latency + Transfer time.

   We will take average values for each of the above components of the access time. Thus,

   Average Seek time = (Maximum Seek Time + Minimum Seek time) / 2 = 10ms

   Maximum Rotational Latency = 1/5400 rpm = 11.11 ms

   Now, for a single disk, average latency is one half the time it takes the disk to do one revolution. i.e. 5.56 ms

   However, if we have more than one disk operating in parallel, and we do not have spindle synchronization, the I/O response time of the striped logical device becomes that of the last one to finish. The performance penalty is degraded by the latency of the last drive to become rotationally ready. The average rotational latency delay in such as system has been derived in the paper[1] as: $\frac{n}{n+1}R$ where R is the maximum rotational latency delay.

   We read 65KB of data in the time that we can traverse a whole track i.e. one revolution time

   Transfer Rate = 65 KB / 11.11 = 5.85 KB/msec

   a) Now for small reads, the RAID operation is identical to read on a single disk. Therefore the service time is 10ms+1/2*11.11ms+512/5.85 KB/msec = 15.6 msec
   b) For small writes, we have to read both the user data as well as the parity data before writing out the new data and parity information to disk. Thus, two disks are read and written for a small read. After the read operation, we wait for a rotation of the disk before we are able the new data can be written. Therefore, the service time for the request is :

      Access time = Seek Time + 2/3* Max. Rotational Latency + Transfer time + 1 Rotation time.

      Since, Transfer time is much smaller compared to the other components, it can be ignored.

Access time = 10 msec + 2/3*11.11 msec + 11.11msec = 28.44 msec.

c) Small read-modify-write operation is similar to the small writes. Therefore, the service time is 28.44 msec.

d) Now, for subsequent large operations, we assume that the user data is striped equally across all disks. Since, each track has size 65KB and there are five data disks in each stripe unit, the stripe size on each disk is 1MB/(65KB*5) = 3.15 tracks per disk

However, as seek distance decreases, settling time becomes a relatively more important component of the seek time. Therefore, we will take an average seek time of 10ms to seek to the first track of the stripe on each disk and then the subsequent track switching operation takes the settling time i.e. the minimum seek time of 4 ms for the 3 tracks.

Therefore, the access time = Seek time + 5/6*Rotational Latency + Transfer time + Settling time

= 10ms + 5/6*11.11ms + 1MB/(5*5.85KB/ms) + 3*4ms= 66.26ms

e) For large writes, the access time is same as for the large read operation i.e. 66.26 ms

f) For large read-modify-writes, the access time is:

= 10 ms + 6/7*11.11ms + 2*1MB/(5*5.85KB/ms) + 6*4ms + rotational time = 124 ms

2. In Problem 1. we have calculated the service times for small and large reads, writes and rmws. We have to now find the service rates for these operations. The service rates are basically the reciprocal of the service time.

3. The formula for the response time T_R is given as :

$$T_R = \frac{T_S}{1 - \frac{Rate_{arr}}{Rate_s}}$$

where T_s is the service time and $Rate_{arr}$ and $Rate_s$ are the arrival and service rates respectively.

We know the small read service time and hence the service rate from Problem 1. We substitute these in the above equation and then plot the curve of the response time versus the Arrival rate. The curve turns out to be a hyperbola. Thus, the performance deteriorates with increasing number of I/O requests. This is due to increased device contention.

4. The MTTDL for a RAID level 5 has been derived using Markov Model and with a backpath for the Repair time as:

$$MTTDL = \frac{(MTTF_{disk})^2}{n(n-1)(MTTR_{disk})}$$

From the problem statement, we have $MTTF_{disk}$ = 300,000 hours and $MTTR_{disk}$ = 24 hours and n =6. Substitute the above values in the equation to obtain the MTTDL.

5. For each state A, we obtain the probability that it will be in state A in time t, is given by: For all incoming edges each having probability ρ_i and all outgoing edges each of probability φ_j is given by: $\bar{P}_A = \sum \rho_i P_i - \sum \phi_j P_A$

   Since there are three working states, we have three differential equations to solve. These are:

$$\dddot{P}_A = 0.6666 P_B - 0.01 P_A$$
$$\dddot{P}_B = 0.01 P_A - 0.6666 P_B - 0.008333 P_B + 0.5 P_C = 0.01 P_A - 0.674933 P_B + 0.5 P_C$$
$$\dddot{P}_B = 0.008333 P_B - 0.007142 P_C - 0.5 P_C = 0.008333 P_B - 0.507142 P_C$$

   The mean time to state F can now be given by solving the equations above as:

   MTTF = - [1 1 1] M^{-1} $\begin{bmatrix} P_A[0] \\ P_B[0] \\ P_C[0] \end{bmatrix}$ where $P_A[0] = 1$ and all other initial probabilities are zero

   and M is the coefficient matrix of the differential equation system above.

$$M = \begin{bmatrix} -0.01 & 0.6666 & 0 \\ 0.01 & -0.67493 & 0.5 \\ 0 & 0.008333 & -0.507142 \end{bmatrix}$$

   and we calculate M^{-1} i.e. the inverse of M as:

$$M^{-1} = \begin{bmatrix} -568133.1 & -568033.1 & -560033.6 \\ -8521.3 & -8521.3 & -8401.3 \\ -140.02 & -140.02 & -140.02 \end{bmatrix}$$

   and obtain the MTTF as 576794.5 units.

6. The data layout functions define, that given the stripe number, on which data disk does it reside, and the offset within the data disk as well as the parity disk and offset within the parity disk.

   The stripe number is a where 0 <= a <= m.r where m is the number of data disks and r is the stripes per data disk.

   ParityDisk(a) = 5 - $\lfloor a/5 \rfloor$ % 6

   DataDisk(a) = a % 5 if a%5 < ParityDisk(a)

   (a+1) % 5 otherwise

   Offset(a) = a / 5

   ParityOffset = a / 5