
SLEC Priority Percent Calculation [WIP]

The math note is an expansion on the combinatorics calculation implemented in the MLEC sim's flat DP mode. The concept is simple - when we have a disk failure, we are interested in how many data on the drive (priority percent) belongs to the disk's priority and then we can calculate the time needed for repair in relation to normal RAID. Knowing the repair time in relation to normal RAID, we can then calculate the MTTDL of network SLEC DP in relation to normal RAID and mathematically calculate the number of nines for network SLEC DP durability.

The notations used in the note is as follows

- r: number of racks
- B: number of disks per rack
- k: number of data chunks
- m: number of parity chunks
- n: stripe width $(k+m)$
- f: total number of failed disk
- f_s: number of failures in the stripe (number of disks that we need to repair)
- f_r: number of failures in the rack
- $\mathcal{CAP}$: capacity of the disk
- $S_{net_W} = S_{net_R} = S_{net}$: speed of network IO
- $S_{disk_W} = S_{disk_R} = S_{disk}$: speed of disk

Discussion on Speed Bottleneck

We can see that due to the unique property of network SLEC DP, out of the $(k+m)$ chunks, there are at most 1 chunk per rack. We are then interested in the bottleneck during repair, whether its network bound, or disk bound, or both. Lets define the following notations

- S_{net}: network speed
- S_{disk}: disk speed
- B: number of drives in the rack
- f_r: number of failures in the same rack

First we start by assuming that $S_{net} = \infty$, then the bottleneck is surely disk IO. We then have **per rack** r/w bandwidth of $(B-f_r)S_{disk}$(we read from all the disks on the rack concurrently). If we then set the network bandwidth back to S_{net}, we can see that if the collective bandwidth for all disks in the rack is smaller than the network bandwidth ($(B-f_r)S_{disk} < S_{net}$), then diskIO would become a bottleneck.

On the otherhand, if the collective disk bandwidth in a rack is greater than the network bandwidth ($(B-f_r)S_{disk} > S_{net}$), then network would become the bottleneck.

Therefore, by moving terms across the inequality, we can characterize the ***per rack bottle neck*** as following

$$\text{bottleneck} = \begin{cases} disk, & \text{if } \frac{S_{net}}{S_{disk}} > B - f_r \\ network, & \text{if } \frac{S_{net}}{S_{disk}} < B - f_r \\ both, & \text{if } \frac{S_{net}}{S_{disk}} = B - f_r \end{cases}$$

And thus the **per rack repair speed**

$$\text{per rack repair speed} = \begin{cases} S_{disk}(B-f_r), & \text{if } \frac{S_{net}}{S_{disk}} > B - f_r \\ S_{net}, & \text{if } \frac{S_{net}}{S_{disk}} < B - f_r \end{cases}$$

Discussion on Parallelism

Due to the constraint that network SLEC DP places on chunk placement, when there is failure, we read from one node per rack for all other racks (k in total). We use the following notation

- f: total number of failed disk

$$\text{read parallelism} = (r-1)B$$

When writing the reconstructed chunk to the *virtual reserved space*, we still want to utilize all of the virutal reserved spaces on all the surviving disks to maximize reconstruction. Therefore,

$$\text{write parallelism} = rB - f$$

Discussion on Priority Percent Calculation

I. When there is a single failure [distinct rack], this means that

- All the affected stripes must not have another drive on the same rack
- The remaining good drives to select affected stripes out of are $(r-1)B$
- The number of failed drive is 1 and has priority of 1
- The priority percent equation then becomes

$$\frac{ncr((r-1)B, n-1) * ncr(1-1, 1-1)}{ncr((r-1)B+1-1, n-1)} = \frac{ncr((r-1)B, n-1)}{ncr((r-1)B, n-1)} = 1$$

We can see that in terms of parallel repair of DP, we can only read/write from and to the disks that do not reside on the same rack. Therefore we will have the parallelism as $(r-1)B$.

The amplification is reading from k chunks and writing to f chunk.

Therefore the time needed for repair calculation is as follows

$$\frac{\mathcal{CAP}(k+f_s)}{S_{net}(r-1)B/f_s}$$

II. When there are two failures [same rack], this means that the **priority 1 stripes**

- All the affected stripes must not have another drive on the same rack
- The remaining good drives to select affected stripes out of are **still** $(r-1)B$ because damaged stripes will have other chunks sitting on other racks, and the damaged chunk be sitting on either one of the failed drive
- The number of failed drive is 2 and both still has priority of 1
- The priority percent equation then becomes

$$\frac{ncr((r-1)B, n-1) * ncr(2-1, 1-1)}{ncr((r-1)B+2-1, n-1)} = \frac{(r-1)B-n+2}{(r-1)B+1}$$

Making an example with 10 racks, 10 drives per rack, and (8+2) config, and there are two failures on the same rack, the priority percent would be

$$\frac{(10-1)*10-10+2}{(10-1)10+1} = \frac{82}{91} \approx 0.901$$

III. When there are two failures [same rack], this means that **priority 2 stripe**

- Basically same as two failures same rack, priority 1 stripe, except priority
- The remaining good drive is $(r-1)B$
- The number of failed drive is 2
- The priority percent equation then becomes

$$\frac{ncr((r-1)B, n-2) * ncr(2-1, 2-1)}{ncr((r-1)B+2-1, n-1)} = \frac{n-1}{(r-1)B+1}$$

Using the same $r=10, B=10, n=10$ example, we have priority percent equals $\frac{9}{91} \approx 0.0989$

IV. When there are two failures [distinct rack], this means that **priority 1 stripe**

- The remaining good drive is still $(r-1)B-1$ because the priority 1 stripe will have all surviving chunks in all racks except the one that contains the failed chunk. The minus 1 is because in one of the rack containing one of the surviving chunk, there is one failed disk that happens to not impact this stripe.
- The number of failed drive is 2
- The priority percent equation then becomes

$$\frac{ncr((r-1)B-1, n-1) * ncr(2-1, 1-1)}{ncr((r-1)B-1+2-1, n-1)} = \frac{(r-1)B-n+1}{(r-1)B}$$

Using the same $r=10, B=10, n=10$ example, we have priority percent equals $\frac{81}{90} = 0.9$

V. When there are two failures [distinct rack], this means that **priority 2 stripe**

- All the stripes with priority 2 have both of the chunks sitting on each of the failed drive residing in two racks. This means that the remaining good drives to select from is $(r-2)B$.
- The number of failed disk is 2
- The priority of the stripes is 2
- The priority percent equation then becomes

$$\frac{ncr((r-2)B, n-2) * ncr(2-1, 2-1)}{ncr((r-2)B+2-1, n-1)} = \frac{n-1}{(r-2)B+1}$$

VI. When there are three failures [distinct rack], this means that **pirority 3 stripe**

- The remaining good drive is $(r-3)B$
- The number of failed disk is 3
- The priority of the stripe is 3
- The priority perent equation then becomes

$$\frac{ncr((r-3)B, n-3) * ncr(3-1, 3-1)}{ncr((r-3)B+3-1, n-1)} = \frac{(n-1)(n-2)}{[(r-3)B+1][(r-3)B+2]}$$

VII. When there are four failures [distinct rack], this means that **priority 4 stripe**

- The remaining good drive is $(r-4)B$
- The number of failed disk is 4
- The priority of the stripe is 4
- The priority percent equation then becomes

$$\frac{ncr((r-4)B, n-4) * ncr(4-1, 4-1)}{ncr((r-4)B+4-1, n-1)} = \frac{(n-1)(n-2)(n-3)}{[(r-4)B+1][(r-4)B+2][(r-4)B+3]}$$

VIII. When there are n failures across n distinct racks, the stripes with priority n

The priority percent calculation should be the following. First we let the number of failures be f

$$\prod_{i=1}^{f-1} \frac{(n-i)}{[(r-f)B+i]}$$