1. Buffering in Uncontrolled Placement of Data Blocks

Assume that a video content is stored on the disk of the video server as shown in the following figure:

Assume it takes 3ms to display one frame. Compute the initial buffering needed before the playback should begin in order to avoid the playback discontinuity.

**Sol**

We first draw a graph that represents the difference between the number of frames transferred and the number of frames displayed at any given time (# frames transferred - # frame displayed). This difference indicates the actual number of frames remaining in the buffer.
As can be seen, at the time between 36 ms and 42 ms the graph falls below zero. In other words, at that period the difference of the number of frames transferred and the number of frames displayed is negative. This is a starvation problem because there is no frame in the buffer to display. To avoid this problem, we can do read-ahead or pre-buffering to store enough data in the buffer before starting playing back the media.

We find the amount of buffering needed by drawing a graph. This graph is still the difference between the number of frames transferred and the number of frames displayed but are lifted up in such a way that its lowest point is at just above zero.

Then, we draw another graph representing the number of frames transferred.
The intersection of the two graphs shows the point where the amount of initial buffering is needed. This is because initially we only transfer data, not display, and later when we start displaying the graph representing the remaining frames in buffer will never drop below 0. In this question, the amount of buffering needed is approximately 1.67 frames or 4 ms.

2. Video Server (Constrained Placement of Data Blocks)

Suppose a video server has a disk that uses constrained placement of video frames. Using the following information, find the maximum number of users that can be supported by this video server and how many block of each user are retrieved in a round given that number of users.

- 1 block contains 4 frames.
- 72 ms playback time for each video block.
- 2 ms separation between any two consecutive blocks
- 12 ms maximum seek time between any two users’ blocks
- 2 ms time to transfer one frame

Sol

From the question, we have:

\[ \beta = \text{average separation time} + \text{time to transfer one block} = 2\text{ms} + 2\text{ms} \times 4 = 10\text{ms} \]

\[ \alpha = \text{seek time between users} + \text{time to transfer one block} = 12\text{ms} + 2\text{ms} \times 4 = 20\text{ms} \]

\[ \gamma = \text{time to display one block} = 72\text{ms} \]

\[ n_{\text{max}} = \left\lfloor \frac{\gamma}{\beta} - 1 \right\rfloor = \left\lfloor \frac{72}{10} - 1 \right\rfloor = 7 \]

\[ k \geq \frac{n^* (\alpha - \beta)}{\gamma - n^* \beta} \]
The maximum number of users that this video server can support is 7 and given that number of users at least 35 blocks of each user are retrieved in a round.
From the question, we have:

\[ \beta = \text{average separation time} + \text{time to transfer one block} \]
\[ = \text{average separation time} + \text{time to transfer one frame} \times \text{(#frames/block)} \]
\[ = 2\text{ms} + 2\text{ms} \times 4 \]
\[ = 10 \text{ms} \]

\[ \alpha = \text{seek time between users} + \text{time to transfer one block} \]
\[ = 12\text{ms} + 2\text{ms} \times 4 \]
\[ = 20 \text{ms} \]

\[ \gamma = \text{time to display one block} \]
\[ = 72 \text{ms} \]

\[ n_{\text{max}} = \left\lfloor \frac{\gamma}{\beta} - 1 \right\rfloor \]
\[ = \left\lfloor \frac{72}{10} - 1 \right\rfloor \]
\[ = 7 \]

\[ k \geq \frac{n^* (\alpha - \beta)}{\gamma - n^* \beta} \]
\[ = \frac{7 \times (20 - 10)}{72 - 7 \times 10} \]
\[ = \frac{7 \times 10}{72 - 70} \]
\[ = 35 \]

The maximum number of users that this video server can support is 7 and given that number of users at least 35 blocks of user are retrieved in a round.