Sustainable Performance Bounds at Receivers for Contiguously Framed Server Streamed Video Data for VCR like Functions for Online Video on Demand

Dr. Rajesh Bhadada
Department of Electronics and Communication,
JNV University, Jodhpur, India

Dr. K. L. Sharma
Department of Electronics and Communication,
JNV University, Jodhpur, India

ABSTRACT

Ideal online video on demand (VoD) system should make available any video at any time with smooth VCR like functions such as fast forward, fast backward during playback. Providing them by unicast streaming (dedicated channels) is expensive and ties up resources in networks causing blocks. Video data are bulky and need high bandwidth for real time playback in networked storage and playback environment. The resource blocking manifolds when multiple users go online. Sustainable performance can be attained by identifying conditions and bounds for networks, servers and the receiver equipment. Modern video data follows framed structures and makes use of relative coding. Data streaming and buffering has to support not merely simple linear playback but also non sequential users VCR like interactions. They were quite simple under electro-mechanical playback/storage technologies such as VCRs, cd/dvd players. Modern information networks use non-linear compression, non sequential storage, distributed data approaches and that too are coupled with random communication delays. Together they pose challenges against interactive VoD.

This paper presents the performance bounds and conditions for achieving VCR like functions for frame based online streamed data from servers, necessary to achieve sustainable performance at receivers.

Keywords

True Video On Demand, Interactive video playback, buffers, bounds, streamed video data.

1. INTRODUCTION

An on line Video On Demand (VOD) service empowers a user to view any time a video of his choice in full or in part. The part viewing implies VCR like interaction such as Fast Forward, Jump Forward, Reverse play etc. Such interactive video playback requires video presentation, user interaction and physical devices perform under real time constraints while maintaining temporal precedence relationships amongst data.

The temporal specifications facilitate data storage and buffered playback of the video. Overall video playback combines instantaneous, sequential, and independent display of heterogeneous data. The video presentation needs coherent presentation of separated (packetised) streams to a user. Thus there exist a bound on the delivery performance, failing which a user could loose the information content or continuity or coherence during presentation.

Computing machines are based on sequential timing. Videos and movies represent truly sequential information, which is captured, recorded (stored) and transferred fully sequentially. During playback video has to conform to matched relations between precedence and exact timing constraints. In addition interactivity also implies support for temporal access control operations. These operations provide the base functionality on which online interactive video system can be built, including the system support for the operations such as Fast Forward (FF), Jump Forward (JF), Fast Backward (REW or FB), Jump backward (JB), Slow Play (SF), Slow Reverse (SB), Pause (PA), Play/Resume and Stop (STOP).

To provide users a quality VOD facility, VCR like functionality is a must. Simple streaming and playback designs under-perform when VCR functionality is introduced. Principle reason being that streaming has been designed as to maintain merely the continuous video playback and do not have provision to rapidly adapt to smooth VCR like functions.

This paper has been organized as follows. Section-2 introduces related published work. Under section-3 background essentials have been summarised for an on line VOD system. And in section-4 the model for VCR like conditions have been analysed and proposed with formal identification of conditions. The paper concludes with section-5.

2. RELATED WORK

Basic issues deal with the physical level, service level and the human interface level in an on line VOD system. At the physical level, data from different media are multiplexed over single physical connections or are arranged in physical storage. The service level deals primarily with streaming necessary for playback, the interaction between the VOD system and the users, the kind of services and facilities/interactions allowed and support for various temporal access control operations such as reverse or fast playback. The human interface level describes the random user interaction with the VOD system such as searching, selection and response the service provider result in. The VOD system components must interpret them and provide an accurate rendition.

Techniques such as periodic/staggered, pyramid, skyscraper broadcasting have been explored [1, 2, 3] with an objective to minimize resources like server and network bandwidth and space. They also aim to optimise client disk space etc. for a given maximum service latency or minimizing the maximum service latency for a given system resource. These techniques made use of special sizing of segments being streamed across multi channels. It was observed [4] that limited user interactivity during playback could be achieved but their performance become inconsistent when interaction functions applied successively.

A server may have to support multiple users for the same video playback at the same time. With such multicast, VCR functions cannot be achieved easily at individual level. Designs such as batching, patching, catching/piggybacking, split and Merge SAM have been proposed [3,5,6] to achieve VCR functions in scheduled multicast system. They all uses multiple channels need switch between channels and that required quick tunability and multi buffering at receiver. However when successive interaction functions are applied, such techniques do not perform significantly better.

These techniques suffer from common major drawbacks (i) that their performance is inversely proportional to the level of interaction in the system. As the user have tendency to use interactive features such as FF, SF, SB, REW etc. in quick succession or combination during playback of longer duration videos, the advantages/savings/gain from these techniques begin to diminish as they do not follow and (ii) they need revisions in existing networks, servers and line interface at receiver.

This paper focuses on identifying the conditions so as to meet user's VCR like requests within users' memory, without altering the simple serial streamed data transfer. Analysis aims to identify that interactive functions can be achieved with proper formulation and planning of stream and buffer. Dependent upon statistics of download of data stream, next section introduces core issues for enforcement in VOD systems.

3. BACKGROUND

3.1 Efficient Play Back at User VOD Player

Video data are sent from a server to the media player by streaming. A user requests the file from a streaming server and interact using protocols having provision for user interaction with the video stream. Streaming allows presentation of the video while continuing data downloading via the network due to large size of video in contrast with ordinary applications, where bulk data are fully downloaded from the network first, before processing by receiver. Current receivers delay playback (about 3-6 seconds) to eliminate network induced jitter. It first buffers the few seconds compressed video data downloaded and then consumes the buffer. The streaming rate x(t) must be greater than or at least equal to the drain rate d(t). It is momentarily less than d(t) when there is packet loss. Internet like public domain networks never commits time and connection duration guarantee [7], resulting x(t) not being constant. System thus runs on the notion that the video is sent and receiver reads them as quickly, as they can and places the video data into the buffer.

After the initial delay, the VOD player reads from its buffer at a rate d(t) and forwards the compressed video for decompression and playback. In the meantime, network has the potential to retransmit lost packets. On the other side, the fill rate x(t) fluctuates with time due to congestion in the network, whose congestion control strategy may lower the transmit rate x(t) after packet loss for some period. This can empty the client buffer (starvation) and results in unwanted pauses during playback of video. The practical conditions in the network govern the rate x(t) which in turn is directly affected by the size of the client buffer.

If this buffer can hold all of the video file, then network can take advantage of using all the instantaneous bandwidth available to the connection, so that x(t) can become much larger than d(t). If x(t) becomes larger than d(t) for longer periods, it results in a large portion of video prefetched into the client buffer preventing starvation. Vice versa, if the client buffer is small, then average x(t) will be around d(t) under better conditions, but still there is a risk of starvation.

3.2 System Timing Support for Data Delivery

Continuity during playback demands data timely and thus data delivery has to be is time bound. Dedicated hardware mechanisms in conventional VCRs provided a constant rate of playback for homogeneous, periodic sequences of time dependent data. Independent physical data paths in the network provide for concurrency in data streams. When such routed data are migrated to computer memories, their storage on disk loose the true characteristics of dedicated physical data path and hence its implied structure as sequential storage. Real time playback needs data to be delivered from storage as per prespecified schedule and maintain it for entire playback duration. Thus additional functions such as random access to the temporal/concurrent data for time-dependent sequential playback are required.

Network performance can be adjudged by delay parameters such as maximum, minimum and average delay in cases of aperiodic data (text, still images). But for continuous data, instantaneous delay variation across individual data elements or packets causing jitter/pauses in playback becomes critical. If multiple but related data are unavailable/fail to continue at the required time instant, they cause *skewed* playback (Fig. 1).

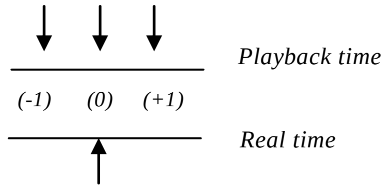


Figure 1: Skew in time

During continuous transfer for video, parts of data can be lost in network, causing frame dropouts or gaps in playback. Such losses cause the stream of subsequent frames to advance in time, resulting in stream lead. Vice versa if a data frame is duplicated, it cause the stream to retard in time and result in stream lag [8].

The ratio of the actual consumption (playback) rate to the available delivery rate of a sequence is termed data utilization and is affected by skew. Data (frame) drops decrease utilization whereas duplicates increase it nominally.

3.3 Handling Synchronization Anomalies

If data gets delayed or unavailable for playback, it results in a gap in the sequence of video at output device TV or Monitor, a synchronization anomaly occurs. Synchronization is challenging to handle/achieve across a network which accommodates multiple independent storage data devices causing random delays, variations in clock rates at each remote data source etc. The delay variation on each channel has to be estimated and buffering need be applied to tackle end-to-end delay, which reshape the channel delay distribution to reduce variance.

Under such variations, a receiver can maintain playback by reconstruction of lost frames [9] or change the playback rate by extension of playback time of previous elements. If gaps (loss of data elements) in a data sequence are ignored with respect to the playback rate, they (gaps) advance the sequence in time when subsequent data are available. This anomaly can be corrected by reduction in playback rate till the schedule catches it again. The expansion method lets each packet be played even if late. The result is the delay of all successive packets and an accumulation of skew with time.

To preserve the duration of the overall sequence, the benefit of the fact that video data has much redundant data (frames) can be taken. Such redundancy offers the scope due to which absence of some data in stream could be ignored. Other ways to handle such gaps is to apply reconstruction based compensation or substitution by alternate data for the missing data in the stream, by the null/non-null/interpolated/ extrapolated data [10,11].

3.4 Impact of Placement of Data Blocks

If data are not available in RAM, the time required to retrieve them depends upon their location over the disk. Thus data blocks distribution can affect the continuous playback performance. To serve FF and FB, more data blocks need be read, virtually meaning additional disk bandwidth. This increase in bandwidth can be achieved by adding more storage devices or adding more RAM, resultant increased cost. Contiguous placement of data blocks effectively utilizes disk bandwidth during normal playback. However an intelligent data retrieval schemes [4,12] can reduce need for extra bandwidth to support interactive functions such as an extra buffer of size nearly several video blocks when a stream is in the FF or FB search modes.

3.5 Managing Efficient Transmission

A channel has to be virtually dedicated for the entire duration of the playback to make system respond to interaction requests immediately. Server and network bandwidths run out rapidly on account of one request one channel allocation. Such unicast is expensive and offers low scalability. Most of the requests are for the videos whose popularity is higher, needing capability to send such video to many users at the same time [13,14]. Multicast facility provides for sharing a server stream among multiple clients to overcome network bandwidth (number of channels) bottleneck. It is also a cause of concern during a live video streaming as how to support the maximum number of users with minimum wait time. Looking to resource limitations if a channel is not available for a request, the request is to be properly handled instead of quick rejection. To be fair, the requests need be served on first-come-first-served basis and users do not wait long if queued. When a request is accepted, a channel need be almost reserved for the entire duration of the video to maintain streaming connectivity.

If there is no queuing of requests, all the accepted requests will achieve zero wait time. But this limits the maximum number of users that can be supported by the system as per the number of channels available for VOD. With this limit even a user demanding the same video being supplied to others may not be accommodated.

Segmentation and multicasting have been proposed to improve the capacity of distribution [12,13]. By segmentation video is divided into fixed length segments which are transmitted at regular intervals in place of transmitting the entire video continuously. For streaming the same (popular) videos at a time to many receivers, multicasting become effective by which same stream is supplied to many users simultaneously using just one or few channels. Techniques such as multiple sender distributed video streaming [15] have been explored to keep check over the bandwidth requirements. However these techniques tend to loose their efficiency and advantages drastically with increased level of interaction allowed to user.

3.6 Download v/s Buffer for Continuous Video

Network conditions and multi-user multi-streaming can not guarantee continuos video stream delivered in a way that the deadlines are easily met without buffering. Further, latencies of storage devices can result in starvation of a video playback. As a result, evaluation of minimum buffering required for a stream that must be allocated and maintained for deterministic playback of single continuous video stream becomes essential [16]. Variable bit rate encoding of video causes effectively variable delays in transmission [17] for same amount of data.

Fig. 2 illustrates the realtionship between the download function $S_D(t)$ and the continuous video consumption function $S_P(t)$. Data are downloaded from the network in packets intermittently starting at time = 0 and end at time = $t_{r.}$ Due to interruptions at the network, no data is downloaded (wherever slope of $S_D(t)$ is 0). During downloads, the slope may vary from one download to the next because of variable compression ratios or network conditions. Once download begins (t=0), after a duration t_0 (minimum one unit buffer), the user begins accessing the data via buffer units for playback. The difference between the download function $S_D(t)$ and the consumption function $S_p(t)$ is the buffer function $B(t, t_0)$ which shows much buffer space is needed vs. time. To prevent playback starvation, the buffer function must be greater than or atleast equal to zero during the interval $t_o \leq t \leq t_r$.

4. VCR LIKE FUNCTIONS MODELING

Simply streaming for sequential video transfer do not have provision for interactions. In this work, the conditions necessary to achieve VCR like functions such as fast forward (FF), Fast Backward (FB/REW) have been modelled for a framed structured video streaming supplied to a limited user buffer. Worldwide currently MPEG is the most followed video compression standard having three types of frames: I, P, and B. It forms independent group of frames; consisting of an I-frame and a certain combination of P and B frames. This modelling is based on video frames received as one unit, to rapidly keep up with changing VCR like actions. A number of frames make some fixed playable length of video defined as segment. This model intends not to consume extra network bandwidth or server resources and does not need alteration in normal sequencing of video data frames. The only limit the model confines to is the small start up latency of receiving one segment. Unbounded functions such as jump forward/jump backward (JF/JB) have been dealt separately. Table-1 enlists the various notations adopted for presentation of this work throughout the subsequent sections.

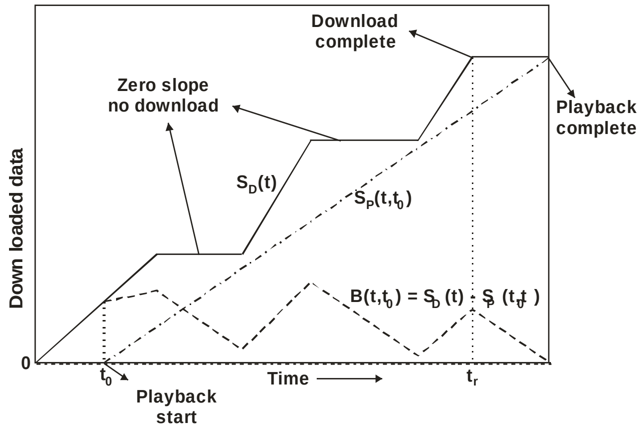


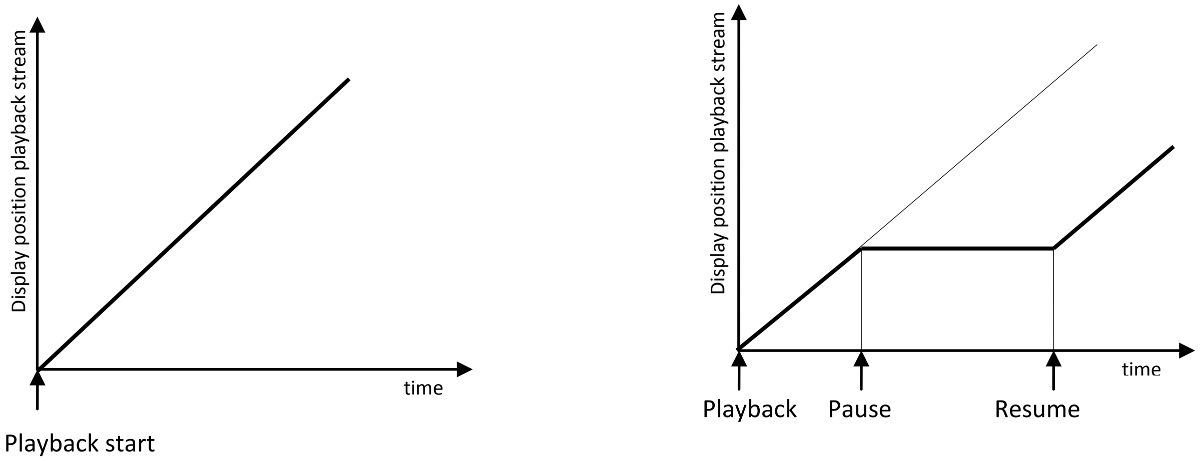
Figure 2: Relationships between download and playback buffer

4.1 VCR functions Categories

Depending upon data position to be played next, various VCR like functions can be naturally categorized as forward or backward except STOP, PA being are directionless. The result of the applied function may maintain continuity in a series of frames displayed or continuity may be broken i.e. temporal notion fails. JF and JB functions are somewhat unbounded and surely break temporal notions. So JF, JB are categorized as *Non-Temporal* functions while all others are *Temporal*. A frame is held over the screen for the pause duration by PA and resume playback with the next frames continuing in temporal order without skipping any next frames. This makes PA as *Temporal* function. Fig. 3 depicts in brief the temporal relations of VCR like functions with respect to normal playback.

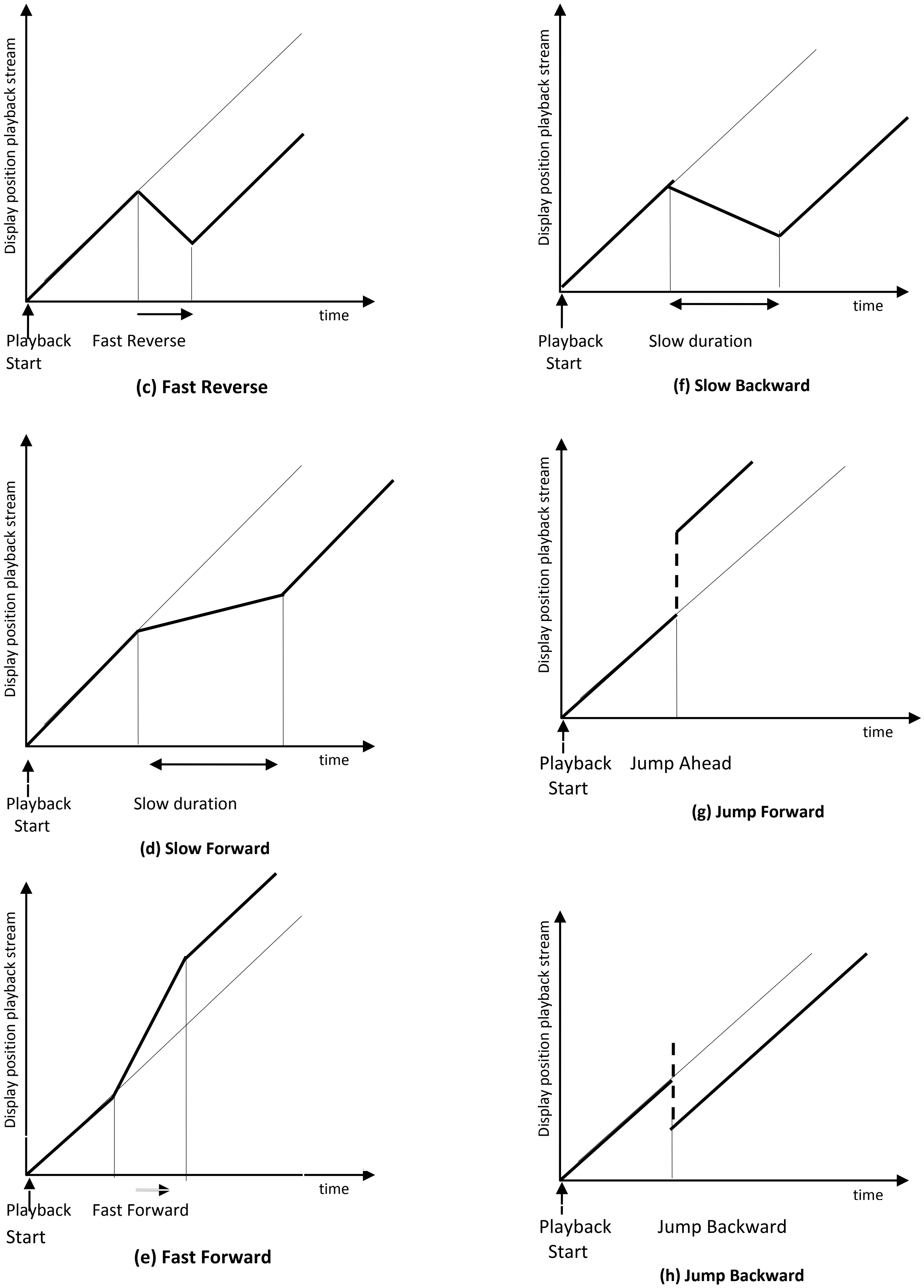
Table 1. Symbol notations adopted

Symbol	Meaning
FF	*Fast forward VCR action while maintaining fast play in forward direction.*
FB/ REW	*Fast reverse or VCR action while maintaining fast play in reverse direction*
PF/PB	*Normal playback in forward or backward direction*
PA	*Pause action of VCR*
STOP	*Stop action of VCR*
JF	*Jump forward action, to jump to any forward position without intermediate display*
JB	*Jump backward action, to jump to any backward position without intermediate display*
SF	*Slow forward VCR action while maintaining slow display in forward direction.*
SB	*Slow reverse VCR action while maintaining slow play in reverse direction*
L	*Total length of video (duration)*
K	*Total number of segments*
Y	*Rate of playback (frames per second)*
B	*Bandwidth of channel for playback rate y*
S	*Segment (number)*
□	*Playback length (duration) of a segment*
J_0	*Current point (playback pointer) / segment / frame*
J	*Target point / segment / frame*
j_b	*Current point in broadcast stream segment / frame*
N	*Number of times (to increase speed for FF/FB)*
R	*User buffer (space for set of segments / frames to be buffered)*
R_f	*Future region of buffer- carries set of segments / frames not yet displayed*
R_p	*Past region of buffer- carries set of segments / frames displayed*
b(j)	*Time left to download a segment / frame from current time*
c(j)	*Time left to consume/display a segment/frame in buffer from current time*



(a) Continuous Playback

(b) PLAY/PAUSE/RESUME


Display position playback stream
time
Playback Start
Fast Reverse
Display position playback stream
time
Playback Start
Slow duration
Display position playback stream
time
Playback Start
Slow duration
Display position playback stream
time
Playback Start
Jump Ahead
Display position playback stream
time
Playback Start
Fast Forward
Display position playback stream
time
Playback Start
Jump Backward

(c) Fast Reverse

(f) Slow Backward

(d) Slow Forward

(g) Jump Forward

(e) Fast Forward

(h) Jump Backward

Figure 3: VCR like interaction operations

4.1 Temporal higher rate stream

The video streams are transferred at higher rates as compared to their actual playback rates, due to the reasons elaborated in the section-3.

4.2 Temporal Functions Conditions

Unordered arrival of data packets causes transfer discontinuities and speed mismatch, making buffering at receiver essential. Buffers make it transparent for the viewer as if data being served by dedicated stream and channel like a unicast VOD. The conditions for providing *Temporal* VCR functions consistently are modeled below and based on them, the minimum buffering requirement is arrived at.

Fig. 4 exhibits a video is virtually streamed as sequence of K segments, being downloaded in the receiver buffer. This buffer has three important regions – the future region, past region and the current position of playback. Playback pointer (j_0) points the video frame under playback currently. Shaded blocks in buffer represent the 'gone' segments (to the left of j_0) those which have been played back while 'coming' segments (to the right of j_0) are those waiting for their playback and they are kept in the 'future' and 'past' regions of buffer. These regions keep shifting with continuous change of playback pointer. The relation in playback point, receive stream point at channel and user buffer position is also illustrated.

i) PF, PB consumes frames as per normal rate (Fig. 3(a)).
ii) STOP does not consume any further buffered frame (Fig. 3(b)).
iii) PA, SF and SB consume frame at lower than normal rate (Fig. 3(b), (d), (f)).
iv) RESUME is as good as PF/PB, i.e. consumes at normal rate (Fig. 3(b)).
v) FF and FB consumes frames at much higher rate (Fig. 3(c), (e)).

Thus if FF, FB actions are met by a model, then remaining (i) to (iv) are bound to meet.

This sub section does not cover discontinuous functions such as JF/JB depicted by fig. 3(g), (h). They have been separately modeled under subsection 4.3.

First the conditions for continuous FF/FB are established as below (refer fig. 4) –

- j_0 is the current point of playback or download, while *j* shows the target point to be reached by the FF action at *n* times the normal playback rate *y*.
- The 'gap' between target point *j* from current point j_0, at normal playback rate *y* can be reached in □ j- $j_0) / y$ time. The player shall consume frames at normal playback □ □ $j, j) =$ □ j- $j_0) / y$.
- For a FF action, this consumption rate is increased by a factor *n; (n > 1).* Then the FF consumption rate shall be □ □ $,j j_0) / n$.
- Thus the consumption time { time to consume all frames from j_0 to *j* denoted by *c(j)* } shall always be greater than □ □ $j, j) / n$, or equal at normal playback ($n = 1$)

  i.e. $c(j) \geq$ □ □ $j, j) / n$ ($n > 1$ for FF).
- If 'future' region carries R_f set of segments (frames), and 'past' region carries R_p set of segments (frames) then $R = R_f \cup R_p$, for all variations across R_f and R_p from 0 to R.
- When j_0 is the current point of playback or download, the target point *j* in the stream shall be having lesser 'gap' as compared to the gap in buffer due to the fact that the buffer will always be lagging behind the transmitted stream.
- To generate real time playback, the download or broadcast rate must be greater than or equal to consumption (playback) rate i.e. at the minimum, the download rate must be at least equal to the playback rate *y*.
- At minimum rate *y*, say time *b(j)* is consumed to reach point *j* in transmitted stream from current j_0.

A continuous FF can be achieved if any segment (frame) not in buffer, gets download before its reduced consumption time (due to FF download) i.e.

□ □ $j, j) / n \geq b(j), \forall j \notin$ R ------***(1)***

$\forall j \in$ R, as $b(j)$=0 for already downloaded segments (frame), it is obvious that $b(j) \leq c(j)$; thus also satisfy

□ □ $j, j) / n \geq b(j)$ ------***(2)***

By equation (1) and (2), it can be concluded that to provide continuous VCR functions, for all 'coming' frames *j*, not yet in buffer; their 'gap' from current play pointer j_0 in the buffer must be $\geq n \cdot b(j)$. Vice versa is also true for 'past' frames.

4.2.1 Minimum Buffer for Temporal Functions

Referring to fig. 4 –

- $s = y \cdot \alpha$ (size of segment = playback rate × playback duration)
- To satisfy eq.(1) derived above, any coming frame *j*, such that □ □ $j, j_0) < n$ be streamed within *c(j)* or within buffer. Thereby in the worst case, $n \times s$ 'coming' frames must be at buffer for *Temporal* FF. Similarly $n \times s$ 'gone' frames must be at buffer for *Temporal* FB (REW).

$\therefore$ At minimum $n \cdot s + n \cdot s = 2\,ns$ buffer size is required for *Temporal* FF/FB actions. ------***(3)***

4.3 Non-Temporal Functions conditions

Operations JF and JB breaks the temporal notion as they require sudden shift playback pointer. This shift might result in two cases – (i) target point lies in the buffer which is simple to deal, mere reloading of value of the playback pointer will suffice or (ii) it shall be out of buffer. This second action is somewhat unbounded (fig. 5) as it renders already buffered content useless. Such unbounded abrupt shift of play pointer can make it unviable to jump to the new destination immediately due to latency in the coming stream for the new position and consequently disrupts even normal playback till the target point in the stream is reached.

Other peculiarity might appear with FB or SB or reverse (REW) functions if they follow an unbounded JF operation. They stand meaningless at the start of playback, but they are true after abrupt jump actions as per new position. To make available such REW facility after a non temporal action, it can only be guaranteed after α seconds or after the nearest α*y* 'gone' segments get buffered. However in cases of unbounded

JB, with the help of storage like hard disk, the JB, REW or reverse actions after jumps can also be provided in short time but still it may not guarantee pauses in playback due to disk latency.

Therefore to tackle the unbounded jumps followings can be undertaken –

1. If no additional resource like bandwidth is to be applied, than instead of waiting for an unbound JF via normal streaming, a request for rescheduling the stream be made to server. To tackle JB, the segments of 'past regions' be backed up in the local storage of receiver.
2. By increasing the bandwidth, create high rate gap between streaming and consumption and support it with a large buffer at receiver, i.e. high bandwidth network for quick download, so that existence of this bottleneck is deferred. It will need extra resources.
3. By multi streaming at multiple channels and thus needing automatically tunable receiver capable of channel switch in short period to catch the future segments early. This shall result in extra channels while only one of them being actually useful.

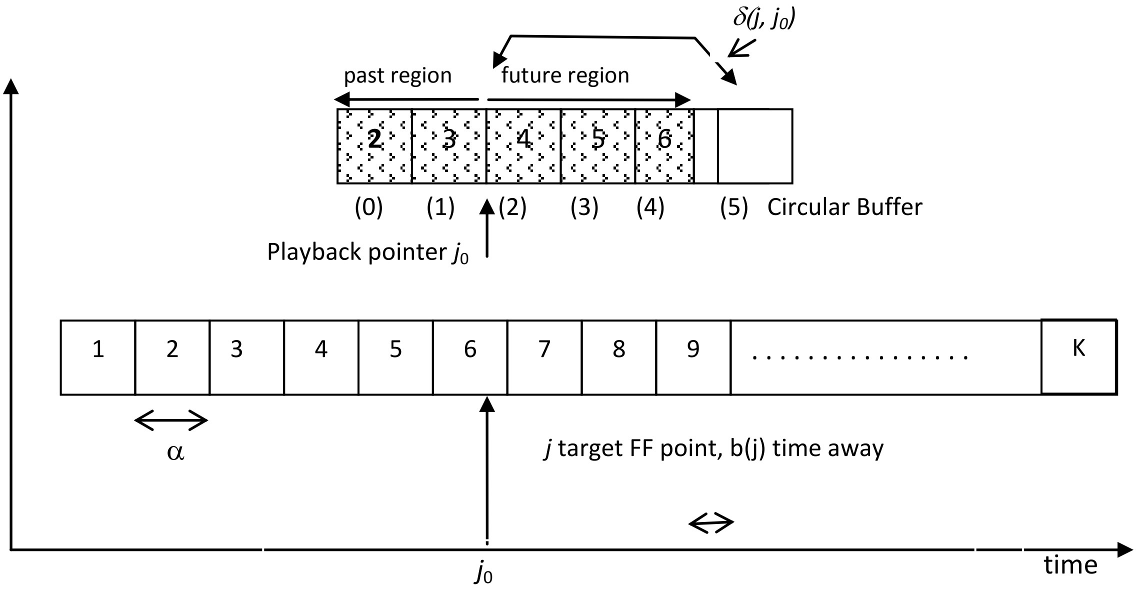


Figure 4 : Regions in buffer and FF target frame and distance

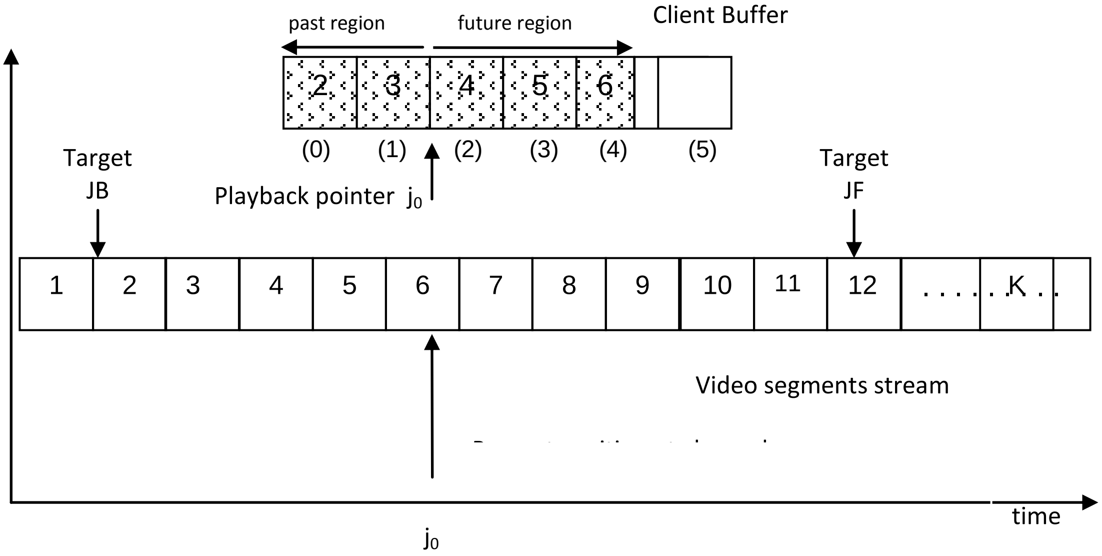


Figure 5: NON TEMPORAL Target jump JF and JB

5. CONCLUSIONS

In this paper, the essential requirements for an on line VOD system providing for VCR like interactions have been summarised. The impact of streaming, its buffering and the factors affecting continuous playback have been reviewed. A model analysing the conditions for successful consideration of VCR like functions at receiver being served by stream of video frames data establishes as under –

(a) *Temporal* functions can be achieved without modifications in simple streaming subject to limits of buffer capacity of receiver and difference in the rate of transmission and consumption.

(b) Partial / Limited *Non-Temporal* functions too can be achieved without modifications subject to limits/bounds of buffer capacity of receiver and difference in the rate of transmission and consumption. All 'gone' frames be stored in receiver disk to accommodate arriving segments while keeping the j_0 nearly around the middle of buffer.

(c) Unbounded *Non-Temporal* functions can be realised better with modifications such as high rate gap between streaming and consumption and larger buffer size at receiver, or by making multi streaming at multiple channels or by ability in the network and server for changing the streaming point rapidly.

6. REFERENCES

[1] Kien A. Hua, Mounir A. Tantaoui and Walla Pak Tavanapong (2004), "Video Delivery Technologies for Large Scale Deployment of Multimedia Applications", Proceedings of the IEEE Vol. 92, No. 9, pp 1439-1448.

[2] S. Ghanehari and L. Ramos (1993), "Continuous retrieval of multimedia data using parallelism", IEEE Transactions on Knowledge and Data Engineering, pp 658-666, Vol. 5, No. 43.

[3] Nabil J Sarhan and Chita R Das (2004), "A New Class of Scheduling Policies for providing time of service guarantees in Video on demand systems", IEE Proceedings on Visual Image Signal Processing, July.

[4] W.F. Poon, K.T.Lo,(2003) "Determination of efficient transmission scheme for video on demand services", IEEE transactions on circuits and systems for video technology, vol 13, pp188-192.

[5] C H Lee and Y B Lee (2003), "Performance analysis of SS-VOD system" IEEE transactions on Circuits and systems for video technology, vol. 12, No. 11, pp 983-997.

[6] Derek Eager, Mary Vernon and John Zahrojan, (2000) "Bandwidth Skimming: A Technique for Cost Effective Video on demand", IEE proceedings of Multimedia Computing and Networking 2000, pp 1-10, San Jose CA.

[7] Paolo Bellavista, Antonio Corradi, (2002) "How to support Internet based distribution of video on demand to portable users", proceedings of 7th IEEE International Symposium on Computers and Communication ISCC '02, IEEE Computer Society, pp 1530-1546.

[8] Phillip Lougher, D. Shepherd and David Pegler, (1994), "The Impact of Digital Audio and Video on High Speed Storage", 13th IEEE Symposium on Mass Storage Systems, pp 84-88.

[9] Jun Peng, Biplab Sikdar, (2005) "An Efficient and scalable loss recovery scheme for video multicast", IEEE Transactions on Multimedia, pp 356-365, April.

[10] Bhadada R. and Rai B., (2009) "Design and performance estimation of Multimedia Information Distribution", Proceedings of International Conference on Emerging and Futuristic System and Technology, pp 150-156, L.I.E.T., Alwar and Institution of Engineers (India), April 09-11, 2009

[11] Chutimet Srinilta, Divyesh Jadav and Alok Choudhary (1997), "Design and Evaluation of Data Storage and Retrieval Strategies in a Distributed Memory Continuous Media Server", IEEE proceedings of the 11th International Parallel Process Symposium (IPPS 97), pp 1063-7133.

[12] Hari Kalva and Borko Furht, (1996) "Techniques for Improving the capacity of video on demand systems", Proceedings of 29th International Conference on Systems Sciences, IEEE, pp 308-315.

[13] Dua T. R., (1998) "Multimedia Infrastructure Requirements Issues, Challenges and Opportunities", Electronics Information and Planning, Vol-25, No. 6, March.

[14] William E. Wright, (2001) "An Efficient Video-On-Demand Model", IEEE Computer, Vol. 34, No. 5, pp 64-70.

[15] Thinh Nguyen and Avideh Zakhor, (2004) "Multiple Sender Distributed Video Streaming", IEEE Transactions on Multimedia, pp 315-326, April.

[16] Bhadada R., (2008) "Study of Efficient Information Management for Video On Demand Systems" Ph.D. thesis under supervision of Dr. Sharma K.L., ECE Dept., JNV Univ. Jodhpur.

[17] W F Poon and K T Lo (2004), "Delay requirement for transmission of VBR videos in interactive broadcast environment", Proceeds of 2nd international conference on Information Technology for Application ICITA, pp68-73.