

## Research Article

# Video Traffic Flow Analysis in Distributed System during Interactive Session

Soumen Kanrar<sup>1,2</sup> and Niranjan Kumar Mandal<sup>2</sup>

<sup>1</sup>*Veher Interactive Pvt. Ltd., Calcutta, West Bengal 700053, India*

<sup>2</sup>*Vidyasagar University, West Bengal 721102, India*

Correspondence should be addressed to Soumen Kanrar; [soumen.kanrar@veheretech.com](mailto:soumen.kanrar@veheretech.com)

Received 25 November 2015; Accepted 15 February 2016

Academic Editor: Martin Reisslein

Copyright © 2016 S. Kanrar and N. K. Mandal. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cost effective, smooth multimedia streaming to the remote customer through the distributed “video on demand” architecture is the most challenging research issue over the decade. The hierarchical system design is used for distributed network to satisfy more requesting users. The distributed hierarchical network system contains all the local and remote storage multimedia servers. The hierarchical network system is used to provide continuous availability of the data stream to the requesting customer. In this work, we propose a novel data stream that handles the methodology for reducing the connection failure and smooth multimedia stream delivery to the remote customer. The proposed session based single-user bandwidth requirement model presents the bandwidth requirement for any interactive session like pause, move slowly, rewind, skip some of the frame, and move fast with some constant number of frames. The proposed session based optimum storage finding algorithm reduces the search hop count towards the remote storage-data server. The modeling and simulation result shows the better impact over the distributed system architecture. This work presents the novel bandwidth requirement model at the interactive session and gives the trade-off in communication and storage costs for different system resource configurations.

## 1. Introduction

An interactive video on demand (VOD) system requires smooth data streaming for the user irrespective of geographic location to access on demand video, such as movies, electronic encyclopedia, interactive games, and educational resources from the distributed storage servers through a high-speed network. The number of customer’s requests increases exponentially in the local cluster domain. It brings a heavy load to the network system. The consequence is a high rate of customer request drop and huge bandwidth wastage. Next-Generation Network (NGN) provides multimedia services over broadband based networks, which supports high definition DVD quality multimedia data streaming content. The multimedia data streaming services are thus seen to be merging mainly in three areas such as computing, communication, and broadcasting. It has numerous advantages. More

exploration for the large-scale deployment of the storage system is still needed. The distributed system hierarchy of multimedia storage has focused on analyzing the system architecture. The distributed system hierarchy of multimedia storage becomes important due to the services supported by the high-speed networks, multimedia storage servers, and distributed multimedia file systems. A customer will be able to request a multimedia stream from anywhere and at any time. In response to a customer’s request, local or remote storage system will deliver a high quality digitized multimedia data stream directly to the clients set-top-box over a local distributed network. The local cluster domain may store a complete or a partial set of multimedia data streams. As the overall VOD user population grows, newer local clusters are added to the distributed system, and the network capacities are sized to match the increased user population for that neighborhood. Clearly, it indicates the

need for scalability of the distributed system hierarchy of a multimedia storage system. The remote site may be archival in nature, providing a permanent repository for all multimedia streams, or they may act as replicated servers such as mirrored sites. The remote sites may provide service to many user populations. The distributed system itself is a hierarchy of neighborhoods in the geographical region. If a request cannot be served from the local site, it may be directed to other remote sites. A storage container either local or remote has to reserve sufficient I/O and network bandwidths before accepting a customer's request. We define a server channel as the server resource required to deliver a multimedia stream, while guaranteeing a client's continuous playback. In general, the VOD service can be characterized as long lived session with a typical movie on demand service runs for 1 hour to 2 hours. Thus, sufficient storage and I/O bandwidth must be available, and we need a solution that efficiently utilizes the server and network resources. Class based admission control [1] was used in the "video on demand system" to get better performance. However, some level of planning and managing of the resource is required in the Internet-based [2, 3] video on demand to enhance the performance up to an optimum level. In fact, the network I/O bottleneck has been observed in many earlier systems, such as Network Project in Orlando [4] and Microsoft's Tiger Video Fileserver [5]. The multicast facilities of modern communication network [6–8] offer an efficient means of one to many data transmissions. Multicast techniques can also significantly improve the VOD performance by reducing the required network bandwidth greatly, so that the overall network load is reduced. In other ways the multicasting alleviates the workload of the VOD server and improves the system throughput by batching requests [9, 10]. In this paper, we present a new methodology to reduce the traffic load in the "distributed video on demand system" for different types of service request in interactive session. The types of the service requests in interactive session are like fast forward, pause, stop, or move backward. Tewari and Kleinrock [11, 12] have designed a simple queueing model to shape the number of replicas corresponding to the request rate and the driven content. Zhou and Xu [13] have shown the target jointly maximizing the average encoding bit rate in content transfer over the network. The overall average number of content replicas that is minimizing the communication load imbalances the storage servers [14]. The brief content placement for P2P video on demand systems is presented by Suh et al. [15]. Tewari and Kleinrock [11, 12] have designed a simple queueing based model to shape the number of replicas corresponding to the request rate of the content driven. LRU (least recently used) and LRFU (least recently and frequently used) based methods worked efficiently and classified the batches of content with respect to the hit count. They efficiently update the peer cache and proxy server cache in the mesh type network for content replicas [9, 10]. The user initiative huge traffic load to the video on demand network generally follows the Zipf like distribution to analyze the traffic load and shape of the load inside the on demand system [6–8]. Li et al. [16] proposed different types of batching request model. This model integrates both the user activity and the batching model. In this model the user requests

are batching and the effect of such batching is captured in a batching model. The user activity includes the various interactive modes like pause, move slowly, move fast, and so forth. The model is unable to describe the user initiative interactive operational mode. The video on demand system requires a uniform model that can be used to determine the requirements of network bandwidth at the user interactive session. The proper bandwidth requirement model at the interactive session gives the trade-off in communication and storage costs for different system resource configurations. This paper is structured as follows. Section 1 presents the brief literature survey for finding the problem in Introduction. We briefly represent the hierarchical architecture of the distributed system and the distributed database storage in Sections 2 and 3. Section 4 presents the session based multiuser model and single-user bandwidth requirement model. Section 5 illustrates the snap of the traffic flow model to this problem. Section 6 presents the parameter's description of the simulation environment and performance evaluation of the system. Section 7 presented broadly the related work on the interactive session and the related domain. The conclusion remark is followed by the references at the end of the document.

## 2. Distributed Architecture

Large-scale VOD system requires the servers to be arranged in a distributed fashion in order to support a large number of concurrent streams. The system architecture is hierarchical, and the local server can handle the requests from specific geographic zone. The local server in the hierarchy takes the requests from the cluster switch; if they cannot handle, then request proceeds for next level. This architecture provides the cost efficiency, reliability, and scalability of servers. Generally, servers are either tree based shaped [9] or graph structured [8]. The graph structured system often offers good quality of service for handling the requests, but the management of request, videos, and streams is complicated in the distributed system. The tree-shaped system can easily manage requests, videos, and streams, but it offers poorer quality of service than the former. In order to evaluate the effectiveness of distribution strategies in such a hierarchy, the authors of [2, 17] investigate how to reduce storage and network costs while taking the customer's behavior into account.

In the distributed system hierarchy of multimedia storage architecture, local proxy servers are installed at strategic locations in the network (closer to the clients). Remote clusters can communicate with the network's distributed storage and with its local proxy servers immediately. Each local cluster server can support a number of customers connected to it through a cluster switch. The customers are connected to the distributed database servers through the cluster switch, which acts as an interface between the client cluster and the database servers. The proxies are used to distribute the storage contained functionality within the network by using the concept of local proxy storage according to Figure 1. If the user cannot be served by the local storage server for any reason, such as a blockage in the

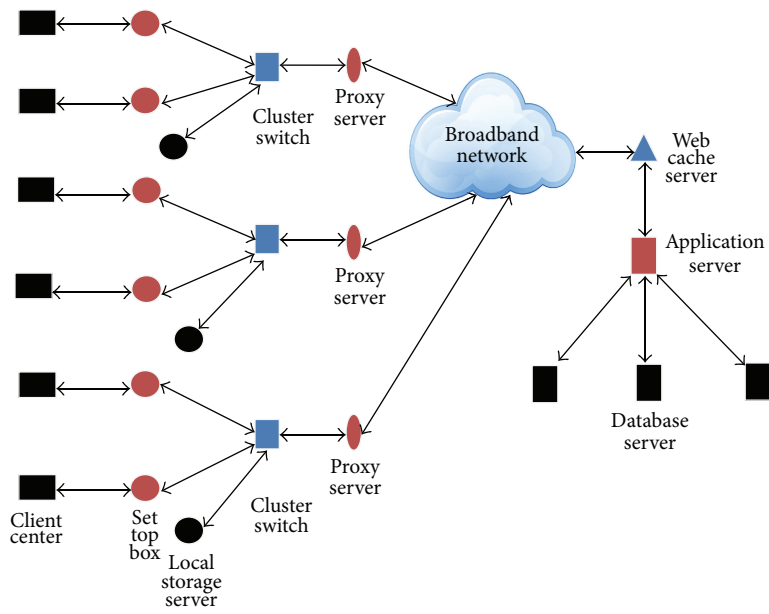


FIGURE 1: Distributed hierarchy system.

local network, or data is not available in the local storage, then the requesting user will be transported to the distributed multimedia storage. By locating the proxy server close to the user, it is expected that there will be significant reductions in the load on the system as a whole [6, 7]. Another advantage of the distributed local proxy video on demand system is that it can be expanded in a horizontal manner for system scalability and evolution. It can start from an initial two-level system (with a centralized multimedia server and one local proxy video server) to a system with as many local proxy servers as needed. The system is incorporated with the distributed multimedia storage system. The distributed system may utilize a lower than average network bandwidth and have a higher system reliability, but at the expense of significant amount of local storage systems.

### 3. Distributed Database Server

If the request is not served by the local storage then it will proceed to remote storage. The working structure of distributed storage is presented in Figure 2. The video on demand viewer sends the request simultaneously via the connected network. Here we present streams of request access from the particular video storage site. Initially, all requests from HTTP or HTTPs are forwarded to web cache memory. Now the web cache memory that has the content of the video stream is forwarded directly to the viewer. In case the web cache memory does not have the required content then the request is forwarded to application server in the list of application servers. The application web server is selected from the least loaded application server [4, 6]. The application web server searches the required video stream from a set of the well connected distributed database servers in Figure 2.

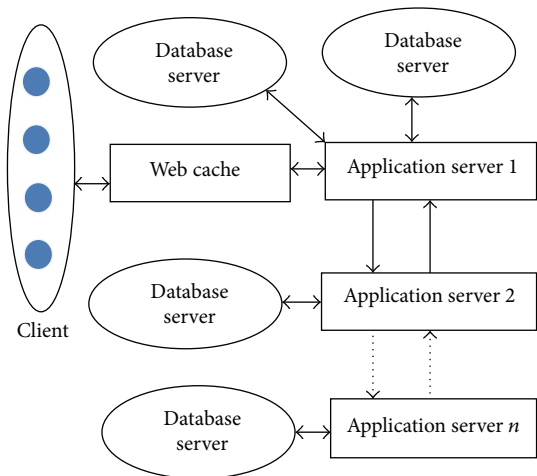


FIGURE 2: Distributed database server.

In the large-scale video on demand system, bandwidth optimization largely depends on the minimum hop counts in high demand stage or in a specific time or some of the days in a week [9]. The efficiency of the video on demand system depends on the “least recently and frequently used” (LRFU) cache updating policy at each server. LRFU policy is composed of the two dynamically growing cache memory blocks “least recently used” (LRU) and “least frequently used” (LFU) cache memory [9]. The performance of the audio video on demand system purely depends on the efficient cache replacement at the proxy server [10]. The initial submitted request pattern to the web proxy container is approximately presented by the Zif like probability distribution [4, 8].

#### 4. Analytic Structure

The required bandwidth varies during the interactive session. Users use various modes of playing operation such as fast forward, skip, pause, and rewind. The required bandwidth continuously varies with the mode of user operation. In general, let  $p$  be the initial (prior) probability to select a channel according to the binomial distribution, the total available bandwidth  $B_{(\Delta t)}$  (i.e., in the trunk) at session  $\Delta t$  and  $W$  is channel required bandwidth at that session, ( $p \approx W/B_{(\Delta t)}$ ). In real time  $W$  (assuming) varies according to the user playing mode of operation within  $(W_{\min}, W_{\max})$  such that  $(W_{\min} \leq \dots \leq W_{\min+t(i)} \leq \dots \leq W_{\max})$  for the  $i$ th type of real interactive session;  $t(i)$  is the bandwidth demand adjustment function. It initially,  $W_{\max}$ , is assigned to each channel, but in real time when a huge request is coming at a session, the channel is assigned with adjusted bandwidth or otherwise at least  $W_{\min}$  is available. If this is impossible, then the request is buffered to the queue (at the 1st queue of the proxy server or any intermediate router) [18]. For connection, setup  $W_{\min}$  is required and then bandwidth is assigned to the channel according to the user's requirement and the availability of the bandwidth. User requirement or required bandwidth is maximized and subject to condition of availability of the bandwidth from the trunk. During the phase of connection startup to connection closed, the user behavior goes through a number of major interactive sessions in a random fashion. We denote the different mode of operations like fast forward ( $I^1$ ), play normally ( $I^2$ ), play slowly or move slowly ( $I^3$ ), rewind ( $I^4$ ), pause ( $I^5$ ), and so forth. So, for the case  $I^i \geq I^j$  it seems to be  $i \geq j$  according to the bandwidth requirement. Now, there is a possibility of more than one mutually exclusive outcome for each type of interactive operation during any session.

Let  $e_i$  for  $1 \leq i \leq k$  be  $k$  mutually exclusive and exhaustive outcome with the respective probabilities  $p_i$  for user mode of operation (*Exp i*).

For example,  $p_1$  is for fast forward,  $p_2$  for play normal,  $p_3$  for play slow,  $p_4$  for rewind, and so forth. If  $e_1$  occurs  $x_1$  times,  $e_2$  occurs  $x_2$  times  $\dots e_k$  occurs  $x_k$  times assuming that  $e_i$  occurs  $x_i$  times for  $1 \leq i \leq k$ .

Now,  $m$  is the number of independent observations of any customer behavior in a session and then  $\sum x_i = m$ .

So for the  $T$ th customer we get

$$P(X = X_T) = p(x_1, x_2, \dots, x_k) = C p_1^{x_1} p_2^{x_2} \dots p_k^{x_k}. \quad (1)$$

Here,  $C$  is the number of permutations of the events.

So,  $C = \frac{m!}{x_1! x_2! x_3! \dots x_k!}$ .

Hence,

$$p(x_1, x_2, \dots, x_k) = \frac{m!}{x_1! x_2! x_3! \dots x_k!} p_1^{x_1} p_2^{x_2} \dots p_k^{x_k} \quad (1)$$

for,  $0 \leq x_i \leq m, \sum x_i \leq m$  (2)

$$= \frac{m!}{\prod_{i=1}^k x_i!} \prod_{i=1}^k p_i^{x_i}.$$

This is true for "single customer" when the allocated bandwidth for that channel is  $\geq W_{\max}$ .

So  $n \approx B_{(\Delta t)}/W_{\max}$  and  $B_{(\Delta t)}$  is the total bandwidth of the trunk in ideal situation or scenario at session  $\Delta t$  (*Exp ii*).

$n$  is the number of active links for corresponding users.

Hence every customer has an independent interactive mode of operational choice.

4.1. *Session Based Multiuser Model.* So, according to function (1) the distribution is

$$P(X_1 \cap X_2 \cap \dots \cap X_n) = \prod_{T=1}^n P(X = X_T). \quad (3)$$

In an interactive session, different bandwidth is required for different service modes like pause, fast forward, rewind, play normally, or play slowly; that is, number of frame downloads by the user node varies, but the frame length is fixed for different interactive session. When the channel is assigned to the user, then at least minimum bandwidth  $y$  is assigned to the user for any interactive session. Consider  $\Rightarrow W_{\min} \leq y \leq W_{\max}$ .

It follows the probability distribution function

$$p_i = \frac{y - W_{\min}}{W_{\max} - W_{\min}}. \quad (4)$$

In a session, we get the mixed type distribution by (1), (2), and (3)

$$P(X_1 \cap X_2 \cap \dots \cap X_n) = \prod_{t=1}^n \frac{m!}{\prod_{i=1}^K x_i!} \prod_{i=1}^K \left[ \int_{W_{\min}}^{W_{\max}} \left[ \frac{y - W_{\min}}{W_{\max} - W_{\min}} \right]^{x_i} dy \right]. \quad (5)$$

Here,  $n$  is the total number of users in the video on demand system.  $m$  is the total number of interactive modes used by the particular customer in that session.  $k$  is the particular type of modes like pause, move forward, rewind, and so forth and  $x_i$  is the number of times in which modes are used by that user in that session.

4.2. *Single-User Bandwidth Requirement Model.* According to Expression (*Exp i*),  $i$  is any type of interactive mode of operation for any individual user. The modes of user operation are like pause, rewind, fast forward, move slowly, and so forth. Let  $b_i$  be the required bandwidth for particular interactive mode of operation in that session,  $W_{\min} \leq b_i \leq W_{\max}$ ,  $b_i > 0$ ,  $i \in I^{>0}$ .

If, in a session, there are possible  $L$  numbers of interactive modes, then the bandwidth requirement of the user mode of operation satisfies expression

$$W_{\min} \leq \max \{b_i\}_{i \in \{1, \dots, L\}} \approx W. \quad (6)$$

In any interactive mode of operation in a session, there are multiple subsessions.

So the occurrence of that particular interactive subsession is

$$P\left(\frac{(X_i = b_i)}{(X_1 = b_1, \dots, X_{i-1} = b_{i-1})}\right). \quad (7)$$

To evaluate (7), we require finding out the volume of the solid sphere in  $k$  dimension plane such that

$$P\left(\frac{(X_i = b_i)}{(X_1 = b_1, \dots, X_{i-1} = b_{i-1}, X_{i+1} = b_{i+1}, \dots, X_k = b_k)}\right). \quad (8)$$

$b_i$  is the required bandwidth to continue and maintain the  $i$ th type active interactive mode and  $c \geq 0$  is the very minimum bandwidth required for other passive interactive user modes of operation at the same session. So the reserved bits for other interactive service connectivity required ( $c \approx W_{\min}$ ).

So expression (8) becomes

$$P\left(\frac{(X_i = b_i)}{(X_1 = c, \dots, X_{i-1} = c, X_{i+1} = c, \dots, X_k = c)}\right). \quad (9)$$

We can assume  $c$  as the required reserved bits for other interactive user modes of operation except  $i$ th type. In general, the  $k$ -dimensional volume of Euclidean ball of radius  $R$  in the  $k$ -dimensional Euclidean space is defined as  $V_k(R) = (\Pi^{k/2}/((k/2) + 1))R^k$  so for even and odd, we get  $V_{2k}(R) = (\Pi^k/k)R^{2k}$  and  $V_{2k+1}(R) = (2^{k+1}\Pi^k/2k + 1)R^{2k+1}$ ; here  $k$  is the number of interactive modes in a session.

Due to interactive mode of user operation at each session, the volume of the sphere is changing. Now we get inequality

$$V_k(R) \approx \frac{\Pi^{k/2}}{((k/2) + 1)} c^{k-1} (W_{\max}) \ll \frac{\Pi^{k/2}}{((k/2) + 1)} (W_{\max})^k. \quad (10)$$

$$P\left(\frac{(X_i = b_i)}{(X_1 = b_1, \dots, X_{i-1} = b_{i-1}, X_{i+1} = b_{i+1}, \dots, X_k = b_k)}\right) \approx \left[\frac{\Pi^{k/2}}{((k/2) + 1)} c^{k-1} (W_{\max})\right]^{-1} \cdot \int_{x_i=0}^{W_{\max}} \int_{x_{i-1}=0}^{[W-(x_1+\dots+x_{i-2}+x_i+\dots+x_k)]} \int_{x_{i+1}=0}^{[W-(x_1+\dots+x_i+x_{i+2}+\dots+x_k)]} \int_{x_k=0}^{[W-(x_1+\dots+x_i+\dots+x_{k-1})]} f(x_1, \dots, x_k) dx_k \dots dx_{i+1} dx_{i-1} dx_i. \quad (13)$$

Here,  $f(x_1, \dots, x_k)$  is a  $k$ -dimensional integrand function related to  $k$ -dimension volume. If the user is active only in one interactive mode then, in special case, we can consider  $f(x_1, \dots, x_k) \approx (1)^k = 1$ .

## 5. Traffic Flow Model

In distributed local proxy architecture, local proxy servers are installed at strategic locations in the network (closer to the clients). Remote viewer clusters can communicate with the networks of distributed primary storage servers and with its local proxy servers. Each local proxy server supports continuous streaming to the number of customers connected to it through a cluster switch. The customers are also connected to the distributed multimedia storage, that is, databases through the cluster switch and proxy server. This composition acts as an interface between the client cluster

Expression (9) becomes

$$P\left(\frac{(X_i = b_i)}{(X_1 = b_1, \dots, X_{i-1} = b_{i-1}, X_{i+1} = b_{i+1}, \dots, X_k = b_k)}\right) \approx \left[\frac{\Pi^{k/2}}{((k/2) + 1)} c^{k-1} (W_{\max})\right]^{-1} \int_{\mathbb{R}^k} dx_1 \dots dx_k. \quad (11)$$

Here we focused on finding the bandwidth requirement for  $i$ th type of user service mode of operation.

Since  $\int_{\mathbb{R}^k} dV = \prod_{i=1}^k (\int_{-\infty}^{\infty} dx_i)$  (integration by parts), allocated bandwidth for that user holds true with the following expression:

$$x_1 + x_2 + \dots + x_k \leq W, \quad (12)$$

$$W_{\min} < x_i \leq W_{\max}, \quad 1 \leq i \leq k.$$

The interactive sessions are mutually independent for each customer; only one type of interactive session occurs, that is,  $i$ th type like pause or watch normally or move forward.

If a viewer opens more than one window, for example, one window is paused and another window is using service required except paused, then the service request will be multiplexed to single stream through that dedicated channel that was allocated to that viewer.

Expression (11) becomes

and the broadband network. The main idea of distributed video on demand (VOD) local storage (as shown in Figure 1) is to distribute the multimedia data functionality within the user cluster by using the concept of local proxy storage. If the streams cannot be served by the local storage due to reasons, such as blockage at the proxy to pull the required segment or chunk of video objects from the local multimedia storage, or the case of the unavailability of the video clips at the proxy server, then the request will be transported to the distributed database servers. To find the proxy server close to the user, it is expected that there will be significant reductions in the load on the system as a whole [6, 7]. Another advantage of the distributed architecture system is that it can be expanded in a horizontal manner for system scalability and evolution. It can start from an initial two-level system (with a centralized multimedia server and one local storage video server) to a system with as many local servers as needed. Compared with the centralized multimedia server system, the



distributed system may utilize a lower than average network bandwidth. It has a higher system reliability, but at the expense of needing a significant amount of local storage to the architecture. The proposed model of the distributed video on demand architecture satisfies the basic requirements of the distributed networks. Let  $n$  be the maximum number of customer's requests, serviceable at the proxy server for a user centric cluster. Now  $R_d$  is the disk bandwidth and  $R_p$  is the client request playback rate including the interactive session at the local storage. Then the stream available at the proxy server will be served to the customer, if the condition still holds true,  $n \leq \lfloor R_d/R_p \rfloor$ . The request admission control test at the proxy server determines whether the proxy server is able to serve maximum  $n$  number of requests. The proposed methodology is used for proxy server. Proxy server has total bandwidth capacity with  $C$  ports. Let us have  $k$  classes of service requests for interactive operations; each class receives mean rate  $\lambda_i$  to the proxy server for  $i = 1$  to  $k$  related to different service classes. Service requests are categories of fast forward with one frame, two frames, up to  $m$  number of frames, pause, move backward, and so forth. Initially the proxy server port capacities are divided into a number of sections with each section having port capacity  $C_j$ , for  $j = 0$  to  $k$ , and  $C_i \neq C_j$  if  $i \neq j$ . The overall proxy port capacity is  $C = \sum_{j=0}^k C_j$ . Initially, each section is assigned with specific category of the service requests. At any interactive session of time  $\Delta t$  the occupied size of the section of ports  $C_j$  for  $j = 0$  to  $k$  can be changed dynamically. It totally depended upon the viewer's choice. So at any session of time  $\Delta t$ , if the request for a particular service class service finds that the specific preassigned section is blocked, then it will search for free port in other sections of the ports. This is mutual shareable port strategy. Request for the section "a" will be admitted to a section "b" with probability

$$p(a, b) = p\left(\frac{(b)}{(a = \text{blocked})}\right) = p_a^b. \quad (14)$$

For example,  $\sum_{i=0}^k p_a^{(a+i) \bmod (k+1)} = 1$ . So, request generates in viewer cluster and arrives at the connected proxy server with mean Poisson rate. Here the maximum ports occupancy for class  $i$  type of service request is  $Q_i$  for  $i = 0$  to  $k$ . When a request for class "a" type of service arrives, check whether  $Q_a > \text{occupancy}(C_a)$ . For simplicity we consider port occupancy at the session  $\Delta t$  is  $C_a^\#$  such that  $\text{occupancy}(C_a) = C_a^\#$  for  $a = 0, 1, \dots, k$ . If so, then a service request of "a" type of service request is admitted to the corresponding "a" section of port with probability  $p_a^a$ . Now,  $p_a^a = (Q_a - C_a^\#)/Q_a$ . So the

request is admitted and updates the port occupancy class as  $C_a^\# = C_a^\# + 1$  at the session  $\Delta t$ . If  $Q_a \leq \text{occupancy}(C_a)$ , then checking continues into other port sections. This happens due to the sectional subnetwork between each proxy to the corresponding local storage in the "distributed video on demand system." This forms a compact topology. To admit the request to the VOD system, checking is continued into the other service section of the port while satisfying the condition  $Q_{a+j} > \text{occupancy}(C_{a+j})$  for finding free port. If free port is available in the other section then the request is admitted by the proxy server and updates the probability  $p_a^{a+j}$  for  $j = 0, 1, \dots, k$  according to Lemma 1 and expression (17).

### 5.1. Lemma

**Lemma 1.** A system will be compact if and only if the execution of any specific event fully depends mutually on other events in compound execution.

*Proof.* In a compact system, the resources are shareable such as channel, link, and port; that is, we prove the converse case; let,  $x_1, x_2, \dots, x_n$  be  $n$  mutually dependent events. Here we show that compound execution  $p(x_1, x_2, x_3, \dots, x_{n-1}, x_n)$  depends on the occurrence of each specific event  $x_i$ , that is,  $p(x_i)$  for  $1 \leq i \leq n$ . Now,

$$\begin{aligned} p(x_1, x_2, x_3, \dots, x_{n-1}, x_n) &= p(x_2, x_3, \dots, x_{n-1}, x_n) \\ &\cdot p\left(\frac{x_1}{(x_2, x_3, \dots, x_{n-1}, x_n)}\right) = p(x_3, \dots, x_{n-1}, x_n) \\ &\cdot p\left(\frac{x_2}{(x_3, \dots, x_{n-1}, x_n)}\right) \\ &\cdot p\left(\frac{x_1}{(x_2, x_3, \dots, x_{n-1}, x_n)}\right) = \dots = p(x_n) \\ &\cdot p\left(\frac{x_{n-1}}{x_n}\right) \\ &\cdot p\left(\frac{x_{n-2}}{(x_n, x_{n-1})}\right) \dots p\left(\frac{x_1}{(x_2, x_3, \dots, x_{n-1}, x_n)}\right). \end{aligned} \quad (15)$$

Hence this proves the system is compact.  $\square$

So according to Lemma 1 and compactness in the local subnetwork of "distributed video on demand system" we get the following expression:

$$\begin{aligned} &p(Q_a \leq C_a^\#, Q_{a+1} \leq C_{a+1}^\#, Q_{a+2} \leq C_{a+2}^\#, \dots, Q_{a+j-1} \leq C_{a+j-1}^\#, Q_{a+j} > C_{a+j}^\#) \\ &= p(Q_{a+1} \leq C_{a+1}^\#, Q_{a+2} \leq C_{a+2}^\#, \dots, Q_{a+j-1} \leq C_{a+j-1}^\#, Q_{a+j} > C_{a+j}^\#) \\ &\cdot p\left(\frac{(Q_a \leq C_a^\#)}{(Q_{a+1} \leq C_{a+1}^\#, Q_{a+2} \leq C_{a+2}^\#, \dots, Q_{a+j-1} \leq C_{a+j-1}^\#, Q_{a+j} > C_{a+j}^\#)}\right) \end{aligned}$$

$$\begin{aligned}
& \vdots \\
& = p(Q_{a+j} > C_{a+j}^{\#}) p\left(\frac{(Q_{a+j-1} \leq C_{a+j-1}^{\#})}{(Q_{a+j} > C_{a+j}^{\#})}\right) \\
& \cdot p\left(\frac{(Q_{a+j-2} \leq C_{a+j-2}^{\#})}{(Q_{a+j-1} \leq C_{a+j-1}^{\#}, Q_{a+j} > C_{a+j}^{\#})}\right) \cdots p\left(\frac{(Q_a \leq C_a^{\#})}{(Q_{a+1} \leq C_{a+1}^{\#}, Q_{a+2} \leq C_{a+2}^{\#}, \dots, Q_{a+j-1} \leq C_{a+j-1}^{\#}, Q_{a+j} > C_{a+j}^{\#})}\right) \\
& = p_a^{a+j} \text{ (say)}.
\end{aligned} \tag{16}$$

By reversing the order we get

$$\begin{aligned}
& p(Q_a \leq C_a^{\#}, Q_{a+1} \leq C_{a+1}^{\#}, Q_{a+2} \leq C_{a+2}^{\#}, \dots, Q_{a+j-1} \\
& \leq C_{a+j-1}^{\#}, Q_{a+j} > C_{a+j}^{\#}) = p_a^{a+j}.
\end{aligned} \tag{17}$$

Now update the occupancy class by  $C_{a+j}^{\#} = C_{a+j}^{\#} + 1$  for any particular  $j = 1, \dots, k$  according to Figure 3. Otherwise, the service request will not be admitted to the local system. If the request cannot be considered, then it is considered blocked for local networking system and the request is to be forwarded to the remote database server before discarding from the distributed system.

The request is forwarded to retrieve video content from the distributed data storage. The distributed data servers are presented in Figure 2. Let us consider  $p_s$  as the probability that  $s$  out of  $N$  video storage servers are active. Active means that the storage server has the required encoded video file, and the end-to-end link is live throughout the streaming at the session  $\Delta t$ . So, we get the expression  $p_s = \binom{N}{s} \rho^s (1 - \rho)^{N-s}$ . The event space is  $\{0, 1\}$  with the probability  $\{1/2, 1/2\}$  for  $\rho$ . Let  $B_{(\Delta t)}$  be the aggregate stationary bit rate from the distributed server according to Figure 2 at session  $\Delta t$ . Now,  $C_{(\Delta t)}$  is the equivalent capacity of networking with respect to bit streaming rate for the whole network according to Figure 1 at session  $\Delta t$ . The value of  $C_{(\Delta t)}$  is obtained by computing the smallest positive integer  $k'$ .  $k'$  is the least positive integer number of servers that have the required video objects or chunk of the video stream. If  $L$  and  $M$  are two events such that  $M = \{B_{(\Delta t)} \leq C_{(\Delta t)}\}$  and  $L = \{\text{At least } k' \text{ number data storage is currently streaming, out of the } N \text{ data storage server at session } \Delta t\}$ ,

$$\begin{aligned}
p\left(\frac{L}{M}\right) &= \frac{p(M/L) p(L)}{p(M)} \\
&= \binom{N}{k'+1} \rho^{k'+1} (1-\rho)^{N-k'-1} + \dots \\
&+ \binom{N}{k'+r} \rho^{k'+r} (1-\rho)^{N-k'-r} + \dots
\end{aligned} \tag{18}$$

for  $r \in I^{>0}$ . So,  $k' \leq N$  is the selected number of minimum database servers from the  $N$  number of database servers.

There is at least  $k'$  number of servers that have the required segment. Now the search for video segments or stream generated a query and broadcast from the application server towards the database servers. So the optimization problem becomes concentrated to find the value of  $k'$ . In the next section, we present the video stream searching procedure to identify the storage location that is the database server IP address.

5.2. *Session Based Query Matched Algorithm.* Algorithm 1 runs at the application server.

## 6. Performance Analyses

The simulation parameters are considered according to the real life streaming scenario in distributed architecture. The parameter's values are summarized and presented in Table 1. For simplicity, we have considered that individual occupancy class of customers is contained in equal number of ports at every "sector of the proxy server." A sector of size  $n$  ports can be occupied by "occupancy class of customer." The size of "occupancy class of customer" is ( $\gg n$ ) during the multithreading operation evoked. The traffic arrival rate with different class of the service requests started from 1Mb/s and ended at 10.5 Mb/s. The number of viewers for each client center is 20. Figures 4–7 present the performance of distributed video on demand system at the local network with respect to analytical methodology.

We have considered 1000 trials and mean of the data is presented in the flowing figures. Figure 4 presents the blockage in the "distributed video on demand system" with respect to the incoming "service requests" traffic flows from the 20 clusters towards the proxy server; here the first sector is blocked. The performance of the distributed system at the local network is evaluated with randomly generated probability set  $p(a, b)$  for the different occupancy class of the service requests. At the first sector, that is, ( $a = 1$ ) and  $2 \leq b \leq 20$ , where sector  $a = 1$  is blocked. Figure 5 presents the performance of the distributed VOD system with respect to the randomly generated probability set  $p(a, b)$  for  $a = 2$ ,  $1 \leq b \leq 20$ , and  $b \neq 2$ ; here sector  $a = 2$  is blocked.

The "request for service" traffic arrival rate to the proxy server varies within 1Mb/second to 10.5 Mb/second with

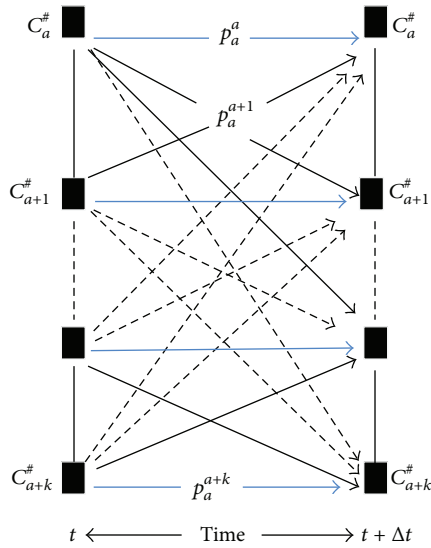


FIGURE 3: Occupancy transition in a session.

increment of 0.5 Mb/second. Figure 6 presents the performance of the distributed video on demand system at the proxy server, with respect to the randomly generated probability set  $p(a, b)$ . Now,  $a = 3$ ,  $1 \leq b \leq 20$ , and  $b \neq 3$ ; here sector  $a = 3$  is blocked. Similarly Figure 7 presents the simulation graph when the sector  $a = 4$  is blocked at the proxy server.

Figures 8 and 9 present the comparative study of system performance at the proxy server during the pull stream operation for different service requirement. The “pull data operation” draws raw data from every local storage to a connected proxy server in distributed architecture. The service requests are different types of service like fast forward, pause, and move backward. Figures 8 and 9 present the comparative graph at one proxy server. The pull operation retrieves multiplexed raw data stream from local storage to the connected proxy server. The plotted graphs are the system performance with and without proposed methodology of the service request handling. Both the figures have shown when the methodology of traffic handling is not used the blocking at the proxy server of “the distributed video on demand system” increases very rapidly. Figure 8 presents the comparative study of a blocked sector  $a = 3$  at the corresponding proxy server. Figure 9 presents the comparative study of a blocked sector  $a = 4$  at the corresponding proxy server. Both Figures 8 and 9 show that the blocking rate increases with an exponential growth without using the methodology.

When we use methodology, the blocking curve is below the threshold level. Clearly the “service of request” can be handled very efficiently by using the methodology at the proxy server for the distributed VOD system. On the other hand, when the load on the video on demand system decreases, this in turn will increase the overall system performance. The packet loss is clearly reduced efficiently by the methodology which enhances the system performance. In the 2nd part of simulation, we consider distributed database storage that is well connected to the application servers.

TABLE 1: Simulation parameters.

Parameter	Value
Number of viewers per client cluster	20
Number of client clusters	10
Different service request arrival (minimum)	1 Mb/sec
Different service request arrival (maximum)	10.5 Mb/sec
Number of ports for each type service	10
Port access time	140 sec
Number of different service classes	20
Number of application servers	7 to 6
Number of database servers	3 to 5
Simulation time	460 sec
<i>Unified parameter (k/n)</i>	
<i>k</i> number of interactive modes used in a session	
<i>n</i> number of active channels used	
Set 1	0.2 to 1.0 step 0.2
Set 2	0.05 to 0.1 step 0.02

The service request is submitted to the proxy server but cannot be served from the connected local server. The service request proceeded further towards the web cache of the distributed database server. The simulation uses the stream search algorithm run at the application server.

In the 2nd stage of simulation, we consider seven application servers, and every application server is well connected with three database storage servers according to Figure 2. Similarly, we consider six application servers, and each application server is connected with five data storage servers in the next phase of simulation. We have considered 90% confidence level for acceptance. From Figure 10, we have noticed that 42000 numbers of user service requests well responded from hop count 4. Figure 11 shows that 51000 numbers of user service requests responded from hop count 3. So the balanced storing distribution of the video stream among the database servers reduces the searching cost and hence reduces the delay and packet drop. If we consider a rough estimate to measure the performance metric for the proposed methodology as the sum of the “number of requests served,” that is,  $l_i$  multiply the hop counts  $h_i$  which gives  $\sum_{i=1}^n (l_i * h_i)$ , the expression gives the following performance metric score value according to Figure 10 that is 368000 and 317000 according to Figure 11. The score value directly depended on the number of computations to search the corresponding stream according to the proposed algorithm. The performance of the system increases as the score value decreases. In the third stage of simulation, we have considered the compact network where the data can be pulled for the user interactive session. The size of the request inside the network grows from 0 to 1000 in one minute, that is, 60 seconds presented in the horizontal axis. The vertical axis is the normalized traffic load with respect to maximum bandwidth on demand requirement, and the distributed network provides the amount at a session. We consider the unified demand parameter as ( $\approx k/n$ ) at 90% confidence



```

(1) At  $t = t_s$  //  $t_s$  is the start time of the
    // session
(2) Var Queue_of_finite_Size : integer
(3) While ( $t \neq (t_s + \Delta t)$ )
(4) {
(5) Initial node {S} // set of data base storage
    // nodes
(6) Boolean matched_node_not_found  $\leftarrow$  True
(7)  $Q \leftarrow$  Queue of finite Size
(8) Begin
(9)  $Q \leftarrow$  push (S) // first in first out
(10) while (match_node_not_found) do
(11) {
(12)      $v \leftarrow$  Pop (Q) // pop is queue operation
(13) For each { $w$ } from single hop neighbor of  $v$ 
(14)     {
(15)     If (query string  $\in w$ )
(16)     then
(17)         {
(18)         match_node_not_found  $\leftarrow$  False
(19)         return node ID of  $w$ 
(20)         exit
(21)         }
(22)     else
(23)      $Q \leftarrow$  Push ( $w$ ) // Push is a queue operation
(24)     } // End for
(25) } // End Inner While
(26)  $t = t_s + \Delta t$ 
(27) } // End Outer While
(28) Send("query matched node not found")
    // To application Server
(29) End
(30) Stop

```

ALGORITHM 1

interval; according to (13) and expression (*Exp ii*),  $k$  is the number of interactive modes used in a session and  $n$  is the number of active links for corresponding users. So,  $n$  depended upon the size of the network at that session. But,  $n \approx B_{(\Delta t)}/W_{\max} \Rightarrow$  parameters  $\approx k(w_{\max})/B_{(\Delta t)}$  and  $B_{(\Delta t)}$  is the total bandwidth of the trunk provided in ideal situation or scenario at session  $\Delta t$ .

Figures 12 and 13 present the traffic load inside the distributed network for various stages of the user driven interactive session. We observed from the figures that the traffic load increases by the increases over the number of users who operates the user driven interactive mode of operation at any session. The unified demand parameter value increases from 0.2 to 1.0 with the step size 0.2 for a particular session. Figure 13 presents the normalized traffic load at that interactive session, for a number of peer nodes, presented in the distributed system.

The peer node is the representation of local storage or proxy servers or distributed storage in distributed system.

Similar type of results is presented in Figure 12 for the unified demand parameter value starting from 0.05 to 0.1 with step size 0.02 for another interactive session with the same set of distributed network environments. In both the cases, we

have considered the duration of the interactive session is 60 seconds and 1000 trials.

## 7. Related Work

Interactive operation on multimedia systems represents a key technology. It is evolving from viewer demand and research prototypes to commercial deployments. Supporting truly and efficiently interactive multimedia services require smooth providing of personalized, dedicated channels or sessions for each user. For this particular set of services, the user requires complete control over the session and has the freedom to explore the depth of both live and stored archive. Development of these interactive multimedia services requires solving a diverse set of technical problems, many of which are tractable. According to Little and Venkatesh, in the interactive services, virtual VCR capabilities and interfaces to view parallel media streams are the most crucial requirement for video on demand system [19]. The earlier work presented by Paxson and Floyd [20], on Poisson processes for analytic simplicity, and a number of traffic studies have shown that packet interarrivals are not exponentially distributed.

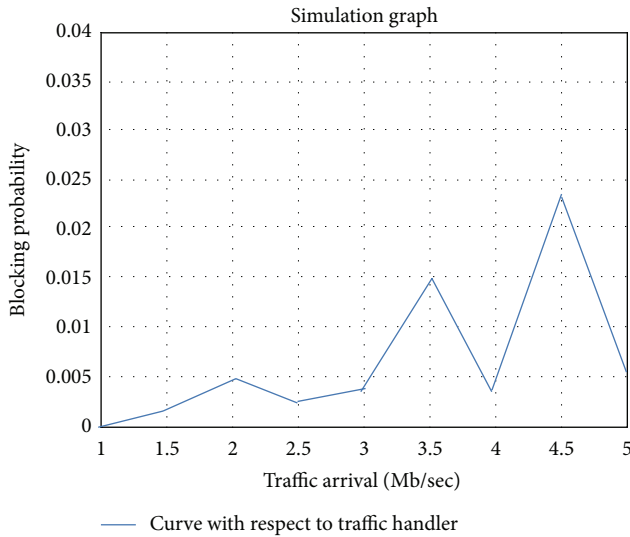


FIGURE 4: Blocking probability versus arrival rate with respect to incoming service requests.

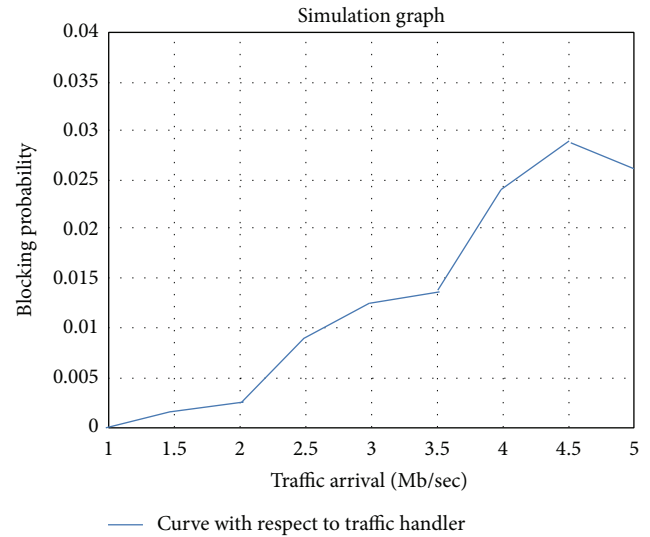


FIGURE 6: Performance with third probability set.

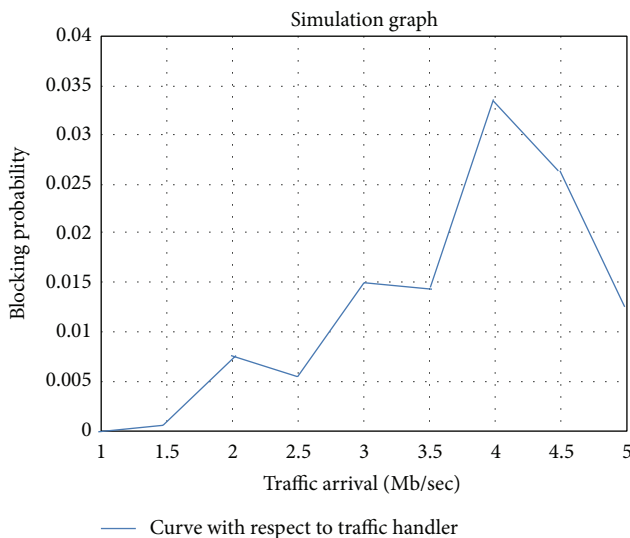


FIGURE 5: Blocking probability versus arrival rate with respect to the randomly generated probability set.

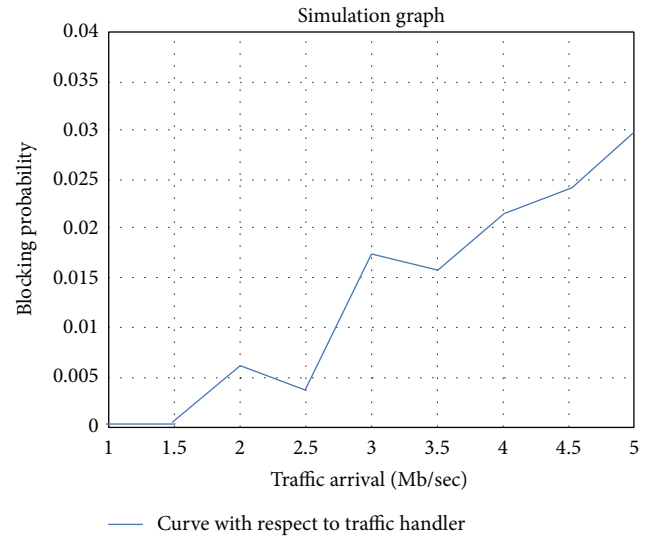


FIGURE 7: Performance with fourth probability set.

In this work, authors considered 24 wide-area traces for investigating a number of wide-area TCP arrival processes. Basically, traces are analyzed: the network, date, duration, and packet (amount). For the analysis purpose tracer considered TCP connection interarrivals, TELNET packet interarrivals, and FTPDATA connection arrivals. The Pareto distribution plays a role both in TELNET packet interarrivals and in the size of FTPDATA burst. Huang et al. [21] mainly focused on the single video stream pull approach. The single peer only redistributes the video stream that it is currently using. This work shows that 95 percentile server bandwidth costs would have been reserved for single peer assisted employment that was instead used. The streaming sessions trace records that were generated by the same Windows Media Player (WMP). Chat application does not consume huge

amounts of traffic. This implies that catering to such users can be a promising way of attracting them, especially in low bandwidth environments. The chat traffic uses neither a well defined port nor a well defined protocol, but there exists a wide spectrum of protocols [22]. It is like IRC protocol used by IRC networks or their applet-based user interface equivalents: custom application protocols used by HTML Web chat systems running on top of HTTP. Diot and Gautier [23] describe the design, implementation, and evaluation of MiMaze. MiMaze is the multiplayer game with a distributed architecture using the multicast backbone for carrying IP multicast traffic over the Internet. It describes the design of dedicated transmission control mechanisms. MiMaze used distributed communication architecture based on the IP multicast protocol suite (RTP/UDP/IP). This work analyzed distributed interactive game on the multicast Internet. Zink

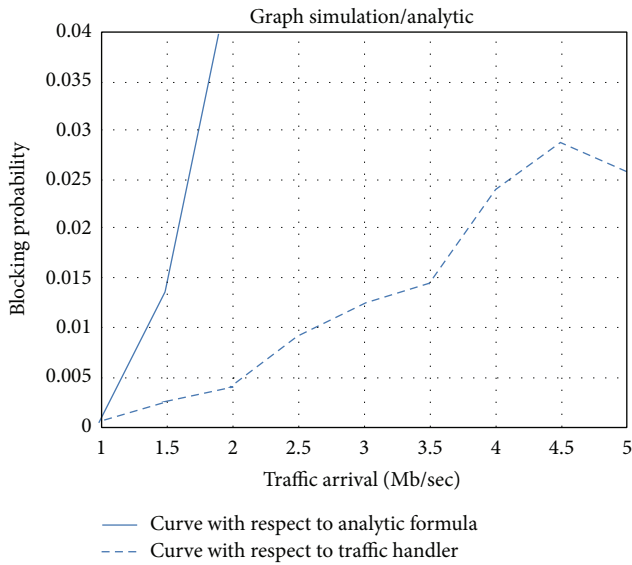


FIGURE 8: Comparison of service request from local storage with third probability set.

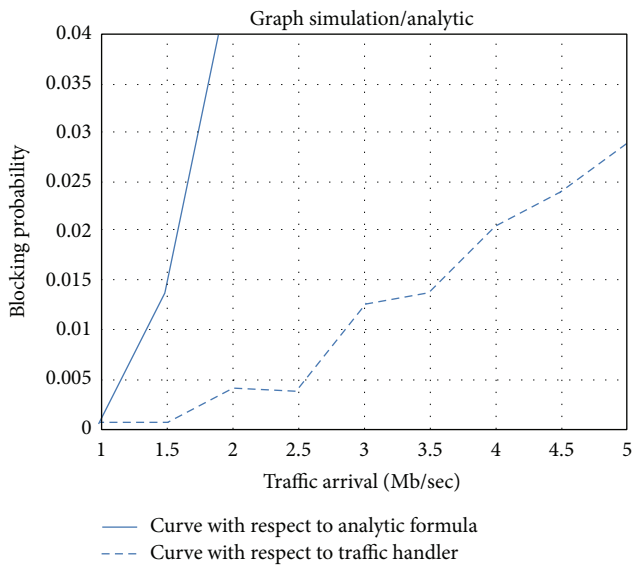


FIGURE 9: Comparison of request server from local storage with fourth probability set.

et al. [24] analyze the content distribution in YouTube. It is realized and conducts a measurement study of YouTube traffic for large university campus network. Based on these measurements, author analyzed the duration and the data rate during streaming sessions. For monitoring the YouTube traffic, a client retrieves a video stream from YouTube. The multithreaded catcher obtains YouTube usage statistics for a particular campus network. Monitoring is done through a two-step process. In the first step, the collected data analyze signaling traffic, that is, HTTP message exchange between the client and the YouTube web server. In the second stage, video streaming is monitored. It is TCP headers of the video data sent from the CDN (content driven network) storage server.

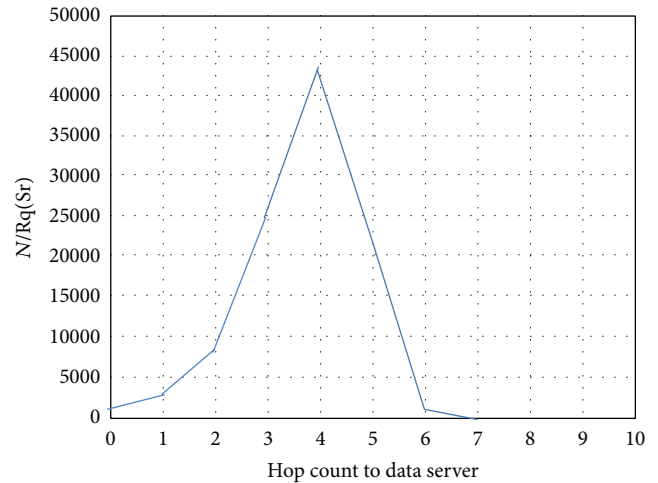


FIGURE 10: Service request served from database server to application server (orientation 1).

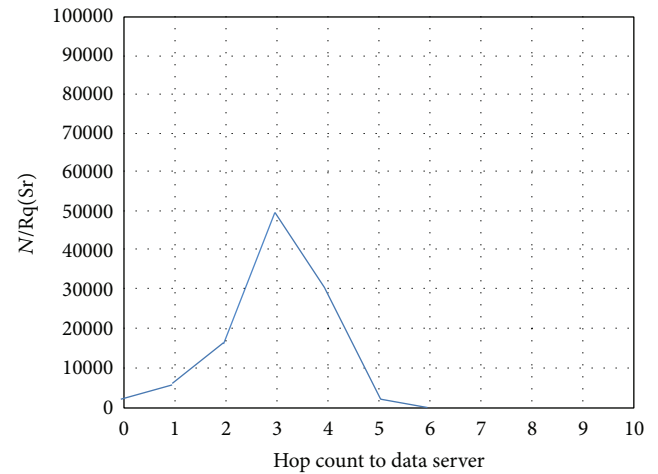


FIGURE 11: Service request served from database server to application server (orientation 2).

The traces are presented as follows: the trace length, user population, and content of local importance on the performance of different distribution architectures. The current tendency in Internet traffic is a shift from web traffic to P2P traffic. Many studies focus on the characterization of P2P traffic analysis [25, 26]. Kim et al. [27] propose a new method to identify current Internet traffic, which is a preliminary but important step toward traffic characterization. The traffic identification method is considered flow grouping in peer-to-peer (P2P) communication architecture. The video traffic is traces by the time-series graph based on the metrics traffic flows, packet, and byte, respectively. Broadband Internet service is popular in many parts of the world. Numbers of research studies have examined the characteristics of such traffic. Maier et al. [28] have considered a common assumption regarding residential traffic. The downstream dominates over the upstream; that is, most bytes are transferred to the local side. Indeed, the considered assumption has shaped and is

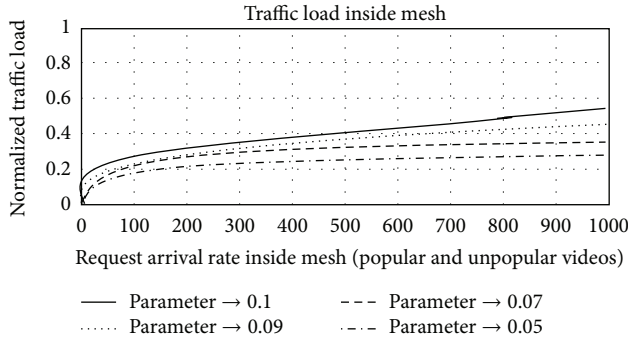


FIGURE 12: Normalized traffic load in interactive session (lower range of parameters).

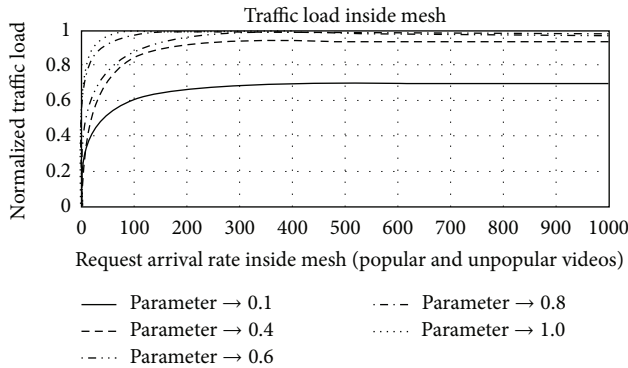


FIGURE 13: Normalized traffic load in interactive session (higher range of parameters).

ingrained in the bandwidth allocations of ADSL (asymmetric digital subscriber line) and cable broadband offerings. In the DSL session characteristics, Maier et al. study with a look at the behavior of the user's DSL sessions (periods of connection to the ISP's network). To trace the traffic, author used HTTP, Bit Torrent, eDonkey, SSL, NNTP, and RTSP protocols.

The performance of traffic monitoring and analysis systems highly depends on the number of flows as well as link utilization and the pattern of packet arrival. Kim et al. [29] examine the characteristics of recent Internet traffic from the perspective of flows. The authors used pipeline based real time traffic monitoring and analysis system architecture. In order to collect IP traffic trace data, authors considered "NG-MON" flow generator simulator and placed at the connection point of the network. The distribution of traffic flow duration and the distribution of the number of packets in flows are presented with the help of TCP and UDP protocols. The simulation result also presented the flow's density. In this work, author used bytes and packets as metrics. In the case of peer-to-peer IPTV communities, Silverston et al. [30] have considered P2P IPTV traffic. It provides useful insights into both transport and packet levels statistical properties of P2P IPTV. It is useful to understand the impact on network nodes and links. The data traffic is collected by peers within the community watching the 2006 FIFA World Cup. The collected data is analyzed on mesh based topological system (PPLive, PPStream, SOPCast, and TVAnts). The tracer plots

the (PPLive, PPStream, SOPCast, and TVAnts) traffic with respect to the number of videos downloaded from the neighboring peers with respect to time. The tracer plots the joint probability density function (PDF) of the interpacket time (IPT) and packet size (PS) of the download traffic. The IPT of each packet is the time that elapsed between that packet and the previous one of the same sessions. Loguinov and Radha [31] analyzed the dynamics of a live streaming experiment that has been conducted between a number of unicast dial-up clients, connecting to the Internet through access points, and a backbone video server. The clients streamed low-bit rate MPEG-4 video sequences from the server over paths with a number of distinct Internet routers. The authors have considered client-server based architecture for MPEG-4 streaming on the Internet. The server was fully multithreaded to ensure that the transmissions of chunk video were performed at the target IP. The bit rate of each streaming session provides a quick response to clients' NACK requests. The streaming was implemented in bursts of packets (with the varying burst duration). The author considered six datasets, each collected from the different machine. The plots are mainly distribution of the number of end-to-end hops and average packet loss rates. The author analyzed the dynamics of a large number of TCP web sessions at a busy Internet server. Liang [32] systematically investigates the long term, online, time variable bit-rate (VBR) video traffic prediction. It is the key and complicated component for advanced predictive dynamic bandwidth control and frame allocation. Since in the Internet and other packet/cell switching broad-band networks (such as ATM), VBR video traffic will be a major part of the traffic produced by multimedia as sources, many researchers have focused on VBR video traffic prediction. Liang [32] traced the prediction of video frame by multiresolution learning neural network (NN) based model.

## 8. Conclusions

In this work, we have presented the three-phase simulation for distributed video on demand system for interactive session. The first stage of simulation according to the proposed methodology reduces the blockage at the proxy server for different types of service requests. It efficiently serves from the local server. If the requested video stream is not present at the local storage or the blockage is above some preassigned threshold then the service request proceeds further to distributed databases. The 2nd stage of the simulation results is presented to reduce the search hop counts for the required storage stream in "distributed video on demand system" by using the session based query search algorithm, which runs at the application server. In the third phase we have seen that the bandwidth requirement completely depended upon the number of interactive modes used by the user at any session. Increase in the number of users brings higher use of interactive modes that leads to the high bandwidth requirement at any session. The size of the network brings little impact to this scenario. The size of the network increases linearly, scalable with the number of peers' nodes in the network. The unified demand parameters do not depend

upon the size of the network or the number of peer nodes. Furthermore, the single-user bandwidth requirement model for interactive mode at any session does not depend upon the number of peer nodes present in the distributed network. The simulation result also reflects the same scenario.

## Competing Interests

The authors Soumen Kanrar and Niranjana Kumar Mandal declare that there is no conflict of interests regarding the publication of the paper.

## Acknowledgments

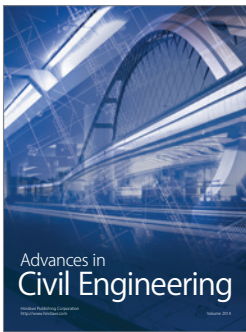
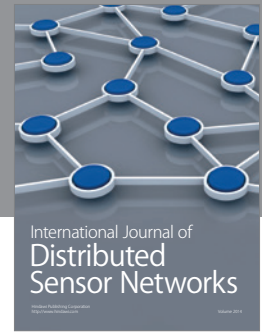
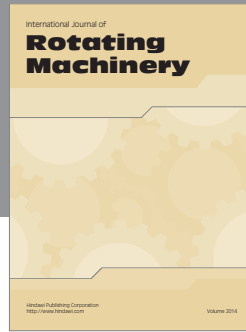
The authors are grateful to Sharmista Das Kanrar from Bishop Westcott, Ranchi, India.

## References

- [1] P. Mundur, A. K. Sood, and R. Simon, "Class-based access control for distributed video-on-demand systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 7, pp. 844–853, 2005.
- [2] S. Annapu Reddy, "Exploring IP/VOD in P2P swarming systems," in *Proceedings of the Conference on Computer Communications (INFOCOM '07)*, pp. 71–75, Anchorage, Alaska, USA, May 2007.
- [3] D. Agrawal, M. S. Beigi, C. Bisdikian, and K.-W. Lee, "Planning and managing the IPTV service deployment," in *Proceedings of the 10th IFIP/IEEE International Symposium on Integrated Network Management (IM '07)*, pp. 353–362, Munich, Germany, May 2007.
- [4] L. Shi, Z. Gu, L. Wei, and Y. Shi, "Quantitative analysis of Zipf's law on web cache," in *Parallel and Distributed Processing and Applications*, vol. 3758 of *Lecture Notes in Computer Science*, pp. 845–852, Springer, 2005.
- [5] G. Hasslinger and O. Hohlfeld, "Efficiency of caches for content distribution on the internet," in *Proceedings of the 22nd International IEEE Teletraffic Congress (ITC '10)*, pp. 1–8, Amsterdam, The Netherlands, September 2010.
- [6] S. Kanrar, "Analysis and implementation of the large scale video-on-demand system," *International Journal of Applied Information Systems*, vol. 1, no. 4, pp. 41–49, 2012.
- [7] S. Kanrar, "Performance of distributed video on demand system for multirate traffic," in *Proceedings of the International Conference on Recent Trends in Information Systems (ReTIS '11)*, pp. 52–56, Kolkata, India, December 2011.
- [8] S. Kanrar, "Efficient traffic control of VoD system," *International Journal of Computer Networks & Communications*, vol. 3, no. 5, pp. 95–106, 2011.
- [9] S. Kanrar and K. N. Mandal, "Dynamic page replacement at the cache memory for the video on demand server," in *Advanced Computing, Networking and Informatics—Volume 2*, pp. 461–469, Springer, 2014.
- [10] S. Kanrar and K. N. Mandal, "performance enhancement for audio-video proxy server," in *Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA), 2014, Volume I*, vol. 327 of *Advances in Intelligent Systems and Computing*, pp. 605–613, Springer, 2015.
- [11] S. Tewari and L. Kleinrock, "On fairness, optimal download performance and proportional replication in peer-to-peer networks," in *Proceedings of the 4th International IFIP-TC6 Networking Conference*, Waterloo, Canada, May 2005.
- [12] S. Tewari and L. Kleinrock, "Proportional replication in peer-to-peer networks," in *Proceedings of the Conference on Computer Communications (INFOCOM '06)*, Barcelona, Spain, 2006.
- [13] X. Zhou and C.-Z. Xu, "Efficient algorithms of video replication and placement on a cluster of streaming servers," *Journal of Network and Computer Applications*, vol. 30, no. 2, pp. 515–540, 2007.
- [14] S. Kanrar and N. K. Mandal, "Optimum storage finding in video on demand system," in *Proceedings of the IEEE 2nd International Conference on Signal Processing and Integrated Networks (SPIN '15)*, pp. 827–830, Noida, India, February 2015.
- [15] K. Suh, C. Diot, J. Kurose et al., "Push-to-peer video-on-demand system: design and evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 9, pp. 1706–1716, 2007.
- [16] V. O. K. Li, W. Liao, X. Qiu, and E. W. M. Wong, "Performance model of interactive video-on-demand systems," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 6, pp. 1099–1109, 1996.
- [17] G. M. Lee, C. S. Lee, W. S. Rhee, and J. K. Choi, "Functional architecture for NGN-based personalized IPTV services," *IEEE Transactions on Broadcasting*, vol. 55, no. 2, 2009.
- [18] A. Dan, D. Sitaram, and P. Shahabuddin, "Scheduling policies for an on demand video server with batching," in *Proceedings of the 2nd ACM International Conference on Multimedia (MULTIMEDIA '94)*, pp. 15–23, San Francisco, Calif, USA, October 1994.
- [19] T. D. C. Little and D. Venkatesh, "Prospects for interactive video-on-demand," *IEEE MultiMedia*, vol. 1, no. 3, pp. 14–24, 1994.
- [20] V. Paxson and S. Floyd, "Wide area traffic: the failure of Poisson modeling," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 226–244, 1995.
- [21] C. Huang, J. Li, and W. Keith, "Can internet video-on-demand be profitable?" *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 133–144, 2007, Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '07).
- [22] C. Dewes, A. Wichmann, and A. Feldmann, "An analysis of internet chat systems," in *Proceedings of the 3rd ACM SIGCOMM Internet Measurement Conference (IMC '03)*, pp. 51–64, October 2003.
- [23] C. Diot and L. Gautier, "A distributed architecture for multi-player interactive applications on the Internet," *IEEE Network*, vol. 13, no. 4, pp. 6–15, 1999.
- [24] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Characteristics of YouTube network traffic at a campus network-measurements, models, and implications," *Computer Networks*, vol. 53, no. 4, pp. 501–514, 2009.
- [25] S. Sen and J. Wang, "Analyzing peer-to-peer traffic across large networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 2, pp. 219–232, 2004.
- [26] S. Saroiu, K. P. Gummadi, R. J. Dunn, S. D. Gribble, and H. M. Levy, "An analysis of internet content delivery systems," in *Proceedings of the 5th Symposium On Operating Systems Design And Implementation ACM SIGOPS Operating Systems Review (OSDI '02)*, vol. 36, pp. 315–327, December 2002.



- [27] M.-S. Kim, Y. J. Won, and J. W.-K. Hong, "Application-level traffic monitoring and an analysis on IP networks," *ETRI Journal*, vol. 27, no. 1, pp. 22–41, 2005.
- [28] G. Maier, A. Feldmann, V. Paxson, and M. Allman, "On dominant characteristics of residential broadband internet traffic," in *Proceedings of the 9th ACM SIGCOMM Internet Measurement Conference (IMC '09)*, pp. 90–102, November 2009.
- [29] M.-S. Kim, Y. J. Won, and J. W. Hong, "Characteristic analysis of internet traffic from the perspective of flows," *Computer Communications*, vol. 29, no. 10, pp. 1639–1652, 2006.
- [30] T. Silverston, O. Fourmaux, A. Bottab et al., "Traffic analysis of peer-to-peer IPTV communities," *Computer Networks*, vol. 53, no. 4, pp. 470–484, 2008.
- [31] D. Loguinov and H. Radha, "End-to-end Internet video traffic dynamics: statistical study and analysis," in *Proceedings of the 21st IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, vol. 2, pp. 723–732, New York, NY, USA, 2002.
- [32] Y. Liang, "Real-time VBR video traffic prediction for dynamic bandwidth allocation," *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 1, pp. 32–47, 2004.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

