Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Computer Networks 47 (2005) 287–306

COMPUTER NETWORKS

www.elsevier.com/locate/comnet

On-demand media streaming to hybrid wired/wireless networks over quasi-geostationary satellite systems

Tarik Taleb *, Nei Kato, Yoshiaki Nemoto

Graduate School of Information Sciences, Tohoku University, Aoba 05, Aza, Aramaki, Aobaku, Sendai 980-8579, Japan

Available online 6 August 2004

Abstract

In order to realize the dream of global personal communications, the integration of terrestrial and satellite communications networks becomes mandatory. In such environments, multimedia applications, such as video-on-demand services, will become more popular. This paper proposes an architecture based on a combination of Quasi-GeoStationary Orbit satellite systems and terrestrial networks for building a large-scale and efficient VoD system.

A hybrid network made of fixed and mobile nodes is considered. The key idea of the architecture is to service fixed nodes according to the neighbors-buffering policy, a recently proposed scheme for VoD delivery, while mobile nodes are served directly from the local server. To allow users to receive their VoD applications with higher degree of mobility, issues related to mobility management are discussed and a simple scheme is proposed to guarantee a smooth streaming of video data.

The importance of the proposed architecture is verified by numerical results. In case of requests coming from fixed nodes within the reach of terrestrial networks, analytical results elucidate the good performance of the architecture in terms of both increasing the system capacity and reducing the disk-bandwidth requirements. Conducted simulations indicate how efficient the proposed system is in smoothening handoffs.

Keywords: Hybrid networks; Mobility management; Video-on-demand; Quasi-geostationary satellite

1. Introduction

New multimedia services, such as video-on-demand, require more cost-effective, high-quality, and high-speed telecommunications technologies. Large-scale deployment of these wide-band services in a metropolitan area with a potentially large

* Corresponding author. Fax: +81 22 263 9306.

E-mail addresses: taleb@nemoto.ecei.tohoku.ac.jp (T. Taleb), kato@it.ecei.tohoku.ac.jp (N. Kato), nemoto@nemoto.ecei.tohoku.ac.jp (Y. Nemoto).

URLs: http://www.nemoto.ecei.tohoku.ac.jp/~taleb (T. Taleb), http://www.it.ecei.tohoku.ac.jp/~kato (N. Kato), http://www.nemoto.ecei.tohoku. ac.jp/~nemoto (Y. Nemoto).

1389-1286/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2004.07.010

number of users, is a challenging task for terrestrial technologies.

Because of their extensive geographic reach, flexible and rapid deployment features, and inherent multicast capabilities, satellite network systems are seen as an attractive solution to realize the vision of a global broadband multimedia infrastructure [1]. Given the recent advances and ongoing improvements in satellite technologies, broadband satellite-based multimedia services are likely to open a promising and strong market for service providers and operators in the near future [2–4].

Whilst satellite systems can play a major role in this broadband multimedia infrastructure, they ought not to perform as an isolated network. They, instead, should be integrated with the existing terrestrial networks and function as a high-speed backbone network to support and/or them backup [5,6]. The design and development of an efficient integration of satellites with terrestrial networks have been thus the subject of extensive research in recent literature and have gained a tremendous interest even at the commercial level [7–9]. Ref. [10] gives a detailed description of cutting-edge techniques and provides updates on the state-of-art prototypes of these satellite–terrestrial hybrid networks.

In this paper, we propose an architecture based on a combination of Quasi-GeoStationary Orbit satellite systems and terrestrial networks for building a significantly large-scale and efficient VoD system. Popular video titles are stored at metropolitan servers and repeatedly transmitted on staggered multicast channels. Individual users in remote areas outside the reach of terrestrial networks are simply serviced via these staggered channels. In case of fixed nodes within the reach of terrestrial networks, if a request comes in between staggered start times of two adjacent multicast channels, the user joins to the most recently started multicast session and then requests the missing part from a nearby neighbor instead of receiving it from a patching unicast channel at a local server. Savings in these unicast channels will be exploited to satisfy requests coming from mobile nodes roaming within the coverage area of the satellite system. To allow users to receive their VoD applications with higher degree of mobility, a simple scheme is proposed to guarantee a smooth streaming of video data during handoff occurrences.

The remainder of this paper is structured as follows. Section 2 gives a brief description of the Quasi-GSO satellite systems and enumerates some of their main merits. Section 3 highlights the relevance of this work to the state-of-art in the context of Video-on-Demand delivery schemes. The key design philosophy and distinct features that were incorporated in the proposed architecture are described in Section 4. Operational details as well as various underlying network assumptions are also discussed in this section. Section 5 presents the three requests admission control policies considered in this paper. Following this, Section 6 considers three types of end-systems, namely fixed nodes within the reach of terrestrial networks, individual users outside the reach of terrestrial networks, and mobile nodes. User requests are handled differently according to the end-system type. In the case of mobile nodes, the section describes a simple scheme to guarantee a smooth streaming of video data during handoff occurrences. Section 7 analytically develops the proposed architecture and describes the simulation philosophy to evaluate the performance of the system in case of handoffs. Analytical and simulation results are presented in the same section. The paper concludes in Section 8 with a summary recapping the main advantages and achievements of the proposed architecture.

2. Quasi-GeoStationary Orbit satellite system

For more than three decades, satellite systems have been successful in providing some commercial services. Currently, there are two types of broadband satellite systems: low-altitude earth orbit and geostationary satellite systems. The former requires a huge infrastructure investment and experiences frequent handover occurrences. The latter, on the other hand, fails to provide a consistently high-elevation angle and consequently experiences frequent incidences of signal propagation cut-off due to tall buildings or mountains.

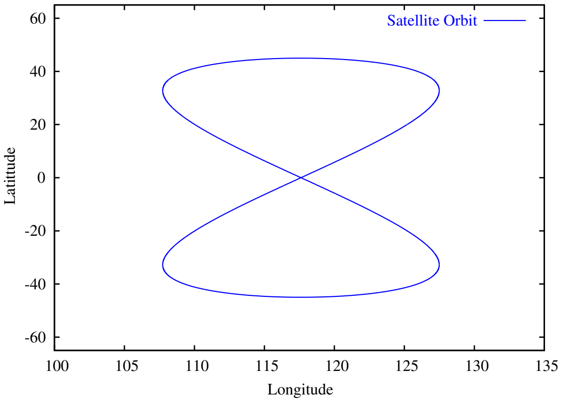

Fig. 1. An example orbit of a Quasi-GSO satellite system made of three satellites.

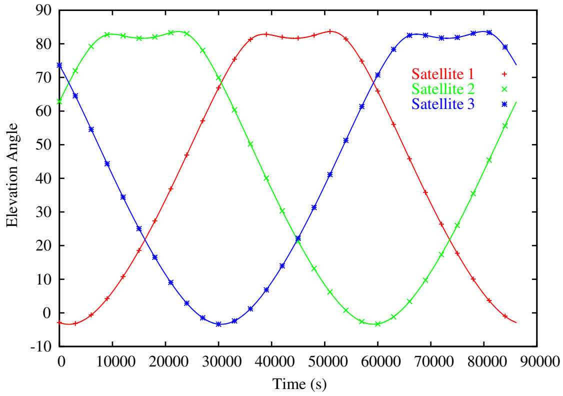

Fig. 2. Elevation angle variation of the three satellites (Tokyo).

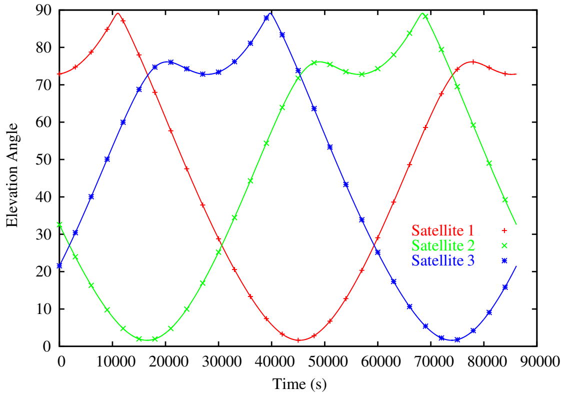

Fig. 3. Elevation angle variation of the three satellites (Sydney).

Needs for a system where satellites have a clear "line of sigh" to the ground, in conjunction with coverage of high latitude regions, have sparked the development of new cost-effective satellite communication systems called Quasi-Geostationary Orbit satellite systems [11].

Quasi-GSO satellite systems provide constant coverage over a particular area of the Earth through employment of a series of satellites. The Quasi-GSO satellites complete one full orbit per day in synchronization with the Earth's rotation, describing a north–south figure of eight locus centered around a point on the equator (Fig. 1). The Quasi-GSO satellite system consists of at least three satellites placed in circular orbits at an inclination angle of approximately 45° relative to the geostationary orbit. The satellites are placed in orbit such that one would be positioned almost directly above the target area at any given point in time. The Quasi-GSO satellites guarantee a minimum angle of elevation of at least 60° and higher values of elevation angle can be achieved by using more than three satellites. Figs. 2 and 3 show the variation of the elevation angles of a Quasi-GSO system made of three satellites at two points on the earth, namely Tokyo and Sydney, respectively. The orbit of the three satellites is as indicated in Fig. 1.

Quasi-GSO satellite systems are a promising alternative to conventional satellites in geostationary or low-altitude orbits. They can deliver huge amounts of broadcasts at high speed with high quality, and without being obstructed by tall buildings. They have been considered efficient for vehicular satellite communications, frequency sharing in fixed satellite communications, positioning systems, and north and south polar regions observation. In addition, they are particularly well suited to the provision of video-on-demand, a service where signal propagation blockings are not tolerated. It should be stressed that the inherent issues with latency of Quasi-GSO satellites do not pose challenges for delivery of high quality multimedia.

This paper aims to study how a combination of these Quasi-GSO satellites and terrestrial networks can result in a powerful tool to distribute bandwidth-intensive multimedia contents, such as video-on-demand, directly to end-users.

3. Video-on-demand related work

In traditional VoD systems, known as True-VoD [12], active users are serviced individually by each being dedicated a video channel during the entire duration of the session. For large-scale networks comprising a potential number of subscribers, the system becomes non-scalable and expensive-to-operate. To improve the scalability and efficiency of large-scale VoD systems, a wide variety of innovative VoD architectures has been proposed in the recent literature.

Batching is one of the leading techniques [13,14]. In this technique, users waiting for the same video feature are grouped and served with a single multicast channel instead of multiple independent unicast channels. While the batching process becomes more efficient as the system scales up, it penalizes earlier clients with a longer wait. To reduce the batching delay, the chaining approach gathers clients from the same batch into a chain [15,16]. The first client of the batch starts playback immediately, caches the video data and then forwards it to upcoming clients in the chain. Another approach for improving VoD systems efficiency is periodic broadcasting [17–19]. In this technique, channels transmitting the same video are offset by a fixed time interval. Video data are available only at the beginning of these slots. A user making a request after the start of a multicast channel should wait till the next upcoming channel starts transmitting the video. The periodic broadcasting approach introduces a significant start-up delay to the customer, which effectively contradicts the on-demand nature of the service. To tackle such an issue, the patching approach allows clients to start playback immediately from a temporary unicast channel while video data from a nearby multicast channel is cached [20,21]. The last approach is piggybacking [22,23]. The basic idea behind this technique is to slightly decrease or increase the playback rate of earlier or upcoming users, respectively, so all users may be served by the same multicast channel.

Attempts on integrating these techniques into hybrid techniques to further improve the efficiency of VoD systems have led to even more sophisticated architectures. Most recently proposed architectures are Unified VoD (UVoD) [24,25], Super Scalar VoD (SS-VoD) [26], and Neighbors-Buffering Based VoD (NBB-VoD) [27,28]. To avoid the long startup delay due to batching, UVoD combines the efficiency of static periodic multicast with the short latency of unicast patching channels by integrating multicast with unicast transmissions. SS-VoD employs dynamic multicast patching channels instead of unicast patching channels in UVOD. This operation ensures that the server will not be overloaded with client requests when the system scales up to a large number of users. In light of the limited resources at any video server, doubts on the performance of the UVoD and SS-VoD approaches may rise when a potentially large number of users issue requests in a short period of time for a certain number of popular video titles. Such cases may happen during the last six evening hours, known as "prime time", when the number of requests for particular popular videos would be rather high [29]. NBB-VoD integrates unicast with multicast transmissions. Additionally, it exploits client-side buffering to satisfy new requests which dramatically improves the system performance at high loads. Given the fact that comparative discussions on the performance of these different approaches is beyond the scope of this study, interested readers are referred to the cited literature.

4. Hybrid network architecture for VoD service delivery

4.1. Key components of the network architecture

The architecture and its components are conceptually depicted in Fig. 4. The figure portrays the coverage area of a Quasi-GSO satellite system. The coverage area is divided into a number of wide service areas, referred to as Metropolitan Service Areas (MSA) throughout this paper. Each MSA area comprises a single metropolitan video-on-demand server and is determined in a way that

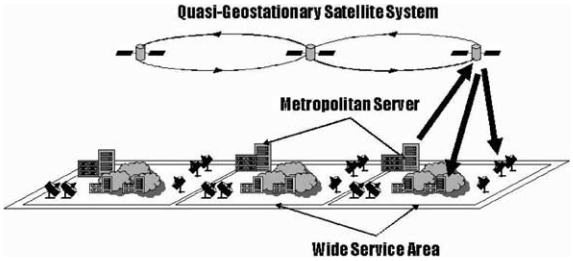

Fig. 4. The coverage area of a Quasi-GSO satellite system divided into a number of Metropolitan Service Areas.

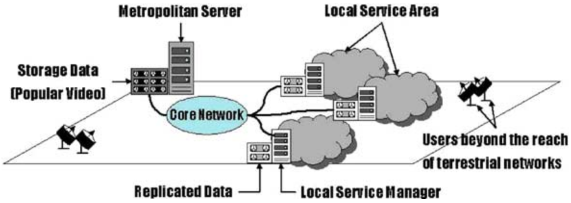

Fig. 5. A Metropolitan Service Area comprising a number of Local Service Areas.

the metropolitan server would always maintain multicast transmissions to end-users with only one hop. This would help to avoid the often unpredictable delay variations and jitter that may be due to handover phenomenon. Obviously, the number of MSA areas should be larger or equal to the number of satellites in the considered Quasi-GSO system. Terrestrial receivers, within a given MSA area, are connected to the correspondent metropolitan server via the Quasi-GSO system. Uplinks refer to transmission from metropolitan video servers to the satellites. Conversely, downlinks refer to multicast transmissions from the satellites to end-users.

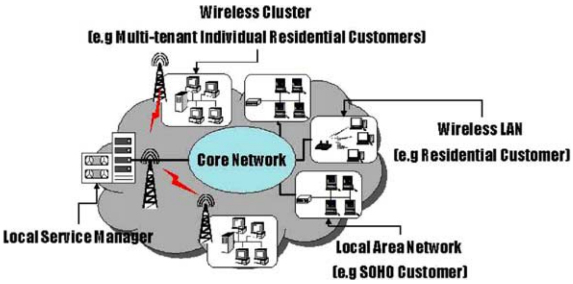

Fig. 6. An example of a Local Service Area.

Fig. 5 shows an example of a MSA area. The illustrated MSA architecture consists of a number of clusters of clients inter-connected via the MSA network backbone. These clusters are referred to as Local Service Areas (LSA) throughout this paper. The MSA area may also include some individual users in remote areas outside the reach of the terrestrial infrastructure.

Fig. 6 depicts a typical example of a Local Service Area. A LSA cluster contains a local VoD service manager and a mini video server. To enable the delivery of digital media to a wide audience and to enhance the consumer experience, the LSA clusters can be thought of as IP transport networks as well. The service manager does authentication and schedules requests for forwarding to the mini video server. The service manager uses information about outstanding requests and the availability of resources to accept or reject requests. Practically, when a request for service arrives, the manager decides whether to deny or accept this request. If the video server can not take additional requests without degrading the quality of service of existing users or causing network congestion, the manager may block the request and send an immediate message to inform the customer that the request has been blocked and that there is need to leave the session. Otherwise, the request will be admitted and the video server will then be requested to allocate a set of available resources to handle the request. The manager should be able to manage a continuous and high throughput, above all, while meeting real-time demands. The mini video server is responsible for processing manager signalings and retrieving adequate data from its storage media. The LSA clusters are formed according to the geographical proximity and the density of end-users. Their determination should be also performed in a way that the mechanisms for accessing and delivering video data are sufficiently fast, reliable, and easy to adapt to users' needs. The LSA network may be made of a hybrid network containing wireless LANs and some LANs inter-connected through the LSA Internet. These wireless LANs or terrestrial LANs can be seen as multi-users platforms, such as corporations, enterprises, small office/home office (SOHO), and residential buildings, where many users are located in the same region and may desire to retrieve the same content over the LSA Internet.

4.2. Video streaming channels

Without loss of generality, all the MSA servers and LSA mini-servers are assumed to be similar. Focus is thus on only one of them. At the MSA server, let there be a total of N_m multicast channels and N_v videos of average length L seconds each. Throughout this paper, a channel is defined as the unit for resource allocation and includes network bandwidth as well as server bandwidth. Videos are assumed to follow a popularity distribution specified by $\{P_k | k = 1, 2, \ldots, N_v\}$ where P_k is the probability of the kth video title to be selected. Whilst estimation of the relative popularity of video titles is in practice difficult, [30] gives an insightful study on the video popularity models and their derivative methods. To assign the multicast channels according to the viewing probability of video titles, the number of multicast channels allocated to the kth video, n_k, is computed as follows:

$$n_k = \frac{\sqrt{P_k} \cdot N_m}{\sum_{j=1}^{N_v} \sqrt{P_j}}. \tag{1}$$

In comparison to a simple uniform channel allocation policy, this channel allocation approach has been analytically proved to further optimize periodic broadcasting system performance [31]. Using a simple staggered multicast schedule, adjacent multicast channels streaming the same video item are offset by a fixed time slot. Depending on the number of multicast channels allocated for a video title, time slots can range from a few minutes to tens of minutes. Assuming the number of multicast channels allocated for the kth video, n_k, to be divisible by the length of the video, L, the time slot W_k, in seconds, is simply

$$W_k = \frac{L}{n_k}. \tag{2}$$

For each multicast channel, the assigned video is repeatedly multicast over the service time regardless of the number of active users or the load of the server, and data transmission from multicast channels is possible at only the beginning of slots. In addition to simplifying the system implementation, this simple staggered schedule also guarantees the support of interactive playback controls without incurring any additional resource allocation or processing at the video servers [32].

At the mini video server, the initial "one slot-time's worth" portions of video titles, contained in the metropolitan server's video library, are replicated. As will be explained later, these initial portions of video data are used to enable clients to start video playback at any time using a unicast channel until they can be merged back onto adequate multicast channels from satellites. Each LSA mini-server has a single request queue shared by a total of N_u unicast channels. The servers serve incoming requests according to the First-Come-First-Serve (FCFS) policy. At the user side, all users' devices are assumed to be similar and capable of concurrently receiving video data from multiple video channels. Additionally, end-systems should have additional local storage to cache up to $\mathrm{Max}_{j=1}^{N_v}(W_j)$ seconds of video data for later playback. For example, 112.5 Mbytes are needed to store 10 min of MPEG-1 compressed video at 1.5 Mbps. This extra buffer can be accommodated using a low cost hard disk on the client side. In case of multiple clients sharing the same local network, this caching can be performed at an intermediate proxy similar in spirit to the idea of [33].

The main advantages behind the considered VoD architectures are threefold. First, metropolitan servers function as periodic broadcasting servers, whereas LSA mini-servers operate independently as True-VoD servers. This modular configuration simplifies the deployment and management of the VoD system. Second, as will be explained later, clients concurrently cache multicasted data at their local buffers and start immediate playback using either a unicast patching from the mini-server or the local buffer of a nearby user. This operation keeps the unicast patchings open for only a short time (say a few minutes), in sharp contrast to True-VoD servers where the unicast channels are occupied for the entire session. This reduction in the service time ultimately increases the system capacity as more requests can be satisfied by the unicast channels. Third, using the multicast channels from the Quasi-GSO system as the main channels for video data delivery to the clients in the wide coverage zone seems advantageous. Firstly, the transmission costs are largely reduced, for they are independent from both the number of clients and its geographical distribution. Secondly, wide area coverage allows communication with only one hop, thus avoiding the often unpredictable delays resulting from routing and congestion in terrestrial networks. Additionally, it eliminates the additional cost that may be incurred by packet duplication and forwarding mechanisms at network routers in terrestrial access subnetworks.

5. Requests admission control mechanisms

This section gives a detailed description of the considered VoD mechanisms. The proposed hybrid integrates ideas from chaining, periodic broadcasting, and patching. Specifically, it combines static multicast channels, unicast patching, and intelligent client-side caching and network bandwidth to reduce the batching delay and vastly increase the servers' capacity.

5.1. Admission via multicast policy

When a user A generates a request at time t_r for a video title, the service manager first checks the start time of the nearest upcoming multicast channel, C_n, transmitting the requested video. Let t_n be this start time. If the waiting time, $(t_n - t_r)$, is smaller than a predetermined admission threshold δ as follows:

$$t_n - t_r \leqslant \delta, \tag{3}$$

the request will be then scheduled for the upcoming channel (Fig. 7). The parameter δ depends on how long the system is willing to let customers wait, and should not be more than few seconds to guarantee short latency service. Recall that a small amount of latency is tolerable to customers because of the relatively long video length. At time t_n, the user A simply joins the upcoming multicast channel and continues receiving video data from the multicast channel.

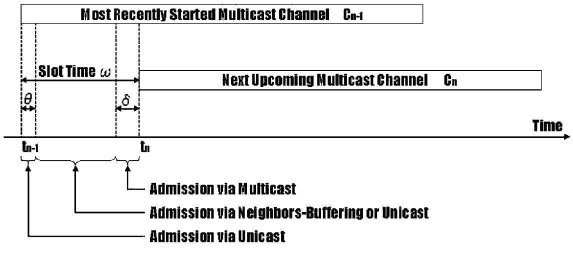
Most Recently Started Multicast Channel C_{n-1}
Slot Time ω
θ
δ
Next Upcoming Multicast Channel C_n
Time
t_{n-1}
t_n
Admission via Multicast
Admission via Neighbors-Buffering or Unicast
Admission via Unicast

Fig. 7. Requests admission control policies.

5.2. Admission via neighbors-buffering policy

This policy is the core idea behind the NBB-VoD approach proposed in [27]. Before delving into a description of the approach, the following definitions should be made. First, a session is formed by having multiple clients receive the same VoD application and is identified by a unique multicast address [34]. A session group is defined as a group of users listening to the same multicast channel. Assuming a frequency division multiplexed system, each session group G_n can be identified by a particular multicast channel C_n. In addition to the assumptions mentioned above, users (to be serviced according to this policy) are assumed to have sufficient bandwidth resources to stream their buffer contents among themselves, and in particular in a secure way that prevents illegal intruders from having any unauthorized access.

Assume a user A issues a request to join a particular session at time t_r in between the staggered start times of two adjacent multicast channels, C_{n-1} and C_n, transmitting video data to the members of the session. Let t_{n-1} and t_n be the start times of the most recently started multicast channel, C_{n-1}, and the nearest upcoming channel, C_n, respectively. If the waiting time, $(t_n - t_r)$, is bigger than the multicast admission threshold δ, the user A will be then requested to join the most recently started multicast channel, C_{n-1}, and store video data from the multicast channel in its local buffer for later playback. As for receiving the missing

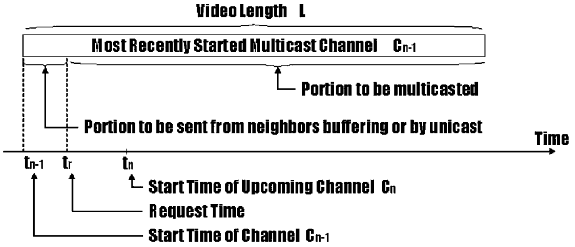
Video Length L
Most Recently Started Multicast Channel Cn-1
Portion to be multicasted
Portion to be sent from neighbors buffering or by unicast
Time
tn-1
tr
tn
Start Time of Upcoming Channel Cn
Request Time
Start Time of Channel Cn-1

Fig. 8. Video portions: one to be multicasted and the other to be sent via unicast channels or nearby buffering.

first portion of the video item, $(t_r - t_{n-1})$'s worth of frames, the user A will be either requested to get it from a nearby user or be assigned a unicast patching channel (Fig. 8).

Since this policy attempts to satisfy new requests by exploiting available resources at neighbors, requests that come in a very short time, θ, after the beginning of the most recently started multicast channel, will be served by unicast channels (Fig. 7). This short time should be set to the maximum time needed by a host in a LSA area to recover from a packet loss [27]. Measurement study of Internet traces shows that even in case of multiple retransmissions, the time required for a host to recover a packet loss, over a terrestrial wide-area network (WAN), is in the order of few seconds. Assume that the issued request comes in a time longer than θ after the beginning of the current multicast channel C_{n-1}

$$t_r - t_{n-1} > \theta. \tag{4}$$

Before the actual reception of the missing part of the requested video data, the user A will be provided with the session's multicast address. The user will then multicast a *session packet* to all the members of the session. In response, each member should send a *reply packet* to the user A. This latter uses these *reply packets* to estimate the one-way distance (in seconds) between the user and other members.

The *session packet* and *reply packets* contain a source-ID and a time-stamp. The time-stamps may be used in a simplified version of the NTP time synchronization algorithm [35]. Assume that user A sends a *session packet* P_s at time t_1 and a user B receives the *session packet* at time t_2. In response to the *session packet*, user B immediately issues a *reply packet* P_r at time t_3 marked with (t_3, Δ) where $\Delta = t_2 - t_1$. Upon receiving P_r at time t_4, user A can estimate the latency from user B to user A as

$$D_{\mathrm{AB}} = \frac{(t_4 - t_3) + \Delta}{2}. \tag{5}$$

Once the distance calculation is done, the user will send a *report packet* to the service manager including information on the one-way distance between the user A and other members. The manager will then use this information to update the *session profile*. Each *session profile* is identified by a name (i.e., video's name). In addition to information about video sequence statistics and users' buffer size, the *session profile* contains the following major elements:

- Client ID—defines a user/member of the session.
- Start-Time—defines the time a user started viewing a video.
- Multicast-Channel—defines the multicast channel a user is listening to.
- Buffer-Contents—indicates the range of frames an end user has in its local buffer.
- Inter-connections—refers to clients that are currently connected to a member.
- Establishment-Time—defines the time an interconnection was established between two users.
- Expiration-Time—defines the time an interconnection will expire.
- Distance—indicates the one-way distance between a user and the other members.

Using the *session profile*, the service manager sorts all the users who have the requested frames within their buffers. If an available neighbor with the appropriate video data is retrieved, an *inter-connection* between the neighbor and user A will be established. The user A will then receive the requested portion of the video item from the neighbor and start an immediate playback. Since user A concurrently caches multicasted video data for the video starting from video time $(t_r - t_{n-1})$, The *inter-connection* between the two users can be released after a time $(t_r - t_{n-1})$ and user A can continue video playback using the locale cache. To guarantee quality of service (QoS), the system designer may set the maximum number of *inter-connections* users' devices can handle simultaneously without any damage to a fixed value. If no user with the desired data is found, the service manager will assign a free unicast stream to the user A to transmit the already-sent part of the video title.

5.3. Admission via unicast policy

If a new user can not be serviced via nearby buffering, the user will then be allocated a free unicast patching channel. Playback should start as soon as video data become available at the user side. At the same time, the user will be required to cache data from the current multicast channel, C_{n-1}. Eventually, playback from the unicast channel will reach the point where the portion of the multicast channel is stored at the local buffer. Henceforward, the unicast channel can be released and the playback is switched to the cached data.

6. Admission control according to the end-system type

In the targeted network environment, three types of end-systems are envisioned, namely individual users in remote areas beyond the reach of terrestrial networks, fixed nodes within the reach of terrestrial networks, and mobile nodes roaming within the coverage area of the satellites. This section explains how each end-system type should be provided with VoD applications and discusses issues related to mobility management.

6.1. Individual users beyond the reach of terrestrial networks

For users in remote areas that are beyond the reach of terrestrial networks, requests are simply satisfied via the Multicast Policy. While this approach reduces the system cost substantially, it obviously affects the on-demand nature of the service due to the delayed server response. To tackle this limitation, similarly to Super Scalar VoD (SS-VoD) [26], the system designer can set aside some dynamic multicast patching channels at the satellites and the metropolitan server to serve users that do not tolerate a startup delay. This option is not considered in this paper.

6.2. Fixed nodes within the reach of terrestrial networks

For this group of nodes, requests are satisfied via the Neighbors-Buffering policy as has been explained earlier. By satisfying the maximum number of new requests, willing to join a particular session, using appropriate buffering of participants in the same session instead of using unicast channels, the effective request arrival rate to the LSA mini-server will be largely reduced. The impact of this reduction on the system is to increase system capacity and make better utilization of available unicast streams. This better utilization can be translated into satisfying requests coming from mobile nodes as will be explained in the remainder of this section.

6.3. Mobile nodes

Employing Quasi-GSO satellite systems for providing VoD services to mobile nodes is in fact an interesting idea. Its interest lies in realizing the dream of a global personal communication system where users are allowed to access VoD applications beyond time and space limitations, and moreover with no signaling blocking because of the high elevation angles of Quasi-GSO systems. Note that these high elevation angles help to avoid stream disruption and drastic quality degradation of stream transmission that may be due to high buildings. In the targeted network environment, the challenges to providing mobile nodes with VoD applications are twofold. First, one needs to decide which admission control policy to use to satisfy requests coming from mobile nodes. Second, to allow users to continue their applications with higher degree of mobility, handoff related issues should be discussed and ways to smoothen handoffs should be investigated.

In admission via neighbors-buffering policy, assignment of a nearby buffer depends on its distance to the new user. In case of Mobile nodes, since they are subject to motion, this distance keeps on changing and updating the session profile upon each change in the distance would lead to a non scalable system. Mobile nodes will thus be satisfied via unicast channels.

Because handoffs are regarded as a key element to guarantee a seamless transmission of VoD applications, they have been placed among top issues in the research of global personal communications. Based on the considered architecture, four types of handoffs are envisioned:

- Intra-LSA handoffs: handoffs between two base stations within a particular LSA.
- Inter-LSA handoffs: handoffs between two adjacent LSAs within a particular MSA.
- Quasi-Inter-LSA handoffs: handoffs between a LSA and a remote area outside the reach of terrestrial networks within a particular MSA.
- Inter-MSA handoffs: handoffs between two adjacent MSAs.

Concerning the Intra-LSA handoffs, their handling can be performed by a set of mobility management techniques that has been proposed in the recent literature. The major objective of most of these techniques is to reduce the packet loss during the handoffs due to the broken data path from the server to the destination. These techniques can be classified into two categories [36]. In the first category, when a handoff occurs, the old base station caches and forwards the packets to the new base station based on a request to forward the packets. Most pioneering examples that use this technique are Fast Handovers Mobile IP [37] and HAWAII [38]. In the second category, packets are routed to multiple nearby base stations around the mobile node to ensure delivery of the packets to the mobile node. In addition to the recently proposed multi-path smooth handoff scheme [39], multicast mobility support [40] and bicast used in Cellular IP [41] use this technique.

Focus is mainly on the three remaining types of handoffs. In Mobile IP, the most dominant protocol among existing mobility management protocols, mobile nodes are identified with two different IP addresses. One is referred to as Home Address and the other is dubbed Care of Address (CoA). The former indicates a unique name of the mobile node and is not subject to change, whereas the latter specifies the position of the node in the network and changes in response to node movement. Upon a handoff occurrence, nodes are assigned different CoAs that should be notified to the Home Agent for binding maintenance.

In the considered architecture, when a user receives video data from a LSA mini-server, the server is assumed to function as the Home Agent of the mobile user. During an Inter-LSA handoff, the mobile user reports its new CoA to the server (HA). Upon reception of the CoA update indication from the mobile IP protocol, the server sends a *profile packet* to the service manager of the adjacent LSA the mobile node has entered. The *profile packet* indicates the remaining video frames the mobile node needs to ensure a continuous playback of the video title. In response to this *profile packet*, the new mini-server (Foreign Agent) who keeps the regional registration for the mobile node opens a unicast channel and starts forwarding the requested frames to the new location of the mobile node. To keep servers always informed of the CoA registrations directly from the mobile nodes, a route optimization option [42] can be used. Note that in Mobile IP, packets destined for the mobile host are intercepted by the HA and tunneled to the FA at the care-of address. The FA then decapsulates the packets and forwards them directly to the mobile host. However, in this approach, the requested packets are directly forwarded from the new LSA mini-server (FA) to mobile nodes.

During handoff, out-of-order and/or duplicate packets may occur. This issue can be resolved by buffering capabilities. In video-on-demand services, a small buffer is typically required to ensure coherent reception, to remove the jitter added by the network, and to recover the original timing relationships between the media data. At the transport layer, mobile nodes are assumed to acquire a small buffer for holding a small number of frames before playing them. This small buffer is responsible for buffering and reordering all the incoming packets. It is also responsible for filtering out the duplicate packets that may occur during handoff before delivering them to the decoder at the application layer. In case of a loss detection, the small buffer should wait for the lost packet for a certain time interval. If the packet does not arrive within the time interval, the buffer delivers its content to the decoder with the missing packets. The time interval should be set in a way that avoids user-level disruption during handoff, and keeps the buffer size and playout delay small. Throughout the paper, this time interval is referred to as playout delay and is denoted as Δ.

In case of a mobile node performing a Quasi- Inter-LSA handoff from a LSA area to a remote area outside the reach of terrestrial network, the LSA service manager should acknowledge the metropolitan server of the handoff and the range of frames the mobile node needs. In response, the MSA server should allocate an emergency channel and start transmitting the requested frames via the Quasi-GSO satellite system to the mobile node. In case of an Inter-MSA handoff, two cases can be envisioned. If the new MSA serves also the video title the mobile node was viewing while being in the old MSA, the handoff will be then treated as either a Quasi-Inter-LSA or Inter-LSA handoff depending on whether the new FA is a LSA or a remote area in the new MSA. If the video title is not serviced at the new MSA, the mobile node will not be able to even cache data of the video title from multicast channels and its request will be denied. Assuming that these two types of handoffs are rare, focus is on the Inter-LSA handoff in the performance evaluation.

7. Performance evaluation

Having described the details of the system, we now direct our focus to evaluating its performance. In the numerical analysis, the number of video items, N_v, and the average length of videos, L, are set to 15 and 90 min, respectively. Unless otherwise specified, δ and θ are set to 60 and 10 s, respectively. It is assumed that all multicast and unicast streams are statistically identical with a transmission capacity C. The request arrival process is assumed to be Poisson with arrival rate λ. This assumption is appropriate because the number of VoD users is typically large and users generate the service requests independently. The viewing probabilities of videos are assumed to follow a normalized geometric distribution. Classifying the N_v videos in order of their popularity, the probability of the ith video to be selected is given then by

$$P_i = \frac{(1-\chi)\chi^{i-1}}{1-\chi^{N_v}} \quad \text{where } i = 1, 2, \ldots, N_v. \tag{6}$$

The parameter χ is called the skew factor. Setting χ to larger values yields a uniform distribution while setting χ to values close to 0 yields highly skewed distribution. In the remainder of this paper, χ is set to 0.7.

7.1. Numerical results: savings in LSA mini-server disk bandwidth

Firstly, it should be emphasized that the following numerical analysis concerns fixed nodes that are within the reach of terrestrial networks and to whom the neighbors-buffering policy is applicable. To contrast the performance of the proposed system to when requests are satisfied via patching unicast channels at the LSA mini-server, a normalized disk-bandwidth reduction factor, Γ, is defined

$$\Gamma = \frac{BW_P^{\mathrm{avg}} - BW_S^{\mathrm{avg}}}{BW_P^{\mathrm{avg}}}, \tag{7}$$

where BW_S^{avg} and BW_P^{avg} are the average disk bandwidth required at the LSA mini-server in case of the proposed system and in case requests are served directly via patching unicast channels, respectively.

We calculate the average disk-bandwidth requirements for users desiring to view a video item k, $\{k = 1, 2, \ldots, N_v\}$. The probability for an incoming user not to be admitted via multicast channels is given by

$$P_{NM} = 1 - \frac{\delta}{W_k}. \tag{8}$$

Assuming the requests arrival process to be Poisson process with arrival rate λ, requests to join the session of the kth video and to be satisfied via Neighbors-Buffering or unicast channels will arrive at a reduced rate

$$\lambda_k = P_k \cdot P_{NM} \cdot \lambda. \tag{9}$$

On the assumption of a Poisson process, the requests inter-arrival times are mutually independent and identically distributed. Request arrivals are assumed to be separated by τ time units. For the sake of presentation clarity, the first request is assumed to arrive τ seconds after the start of the most recent channel (Fig. 9).

When requests for the kth video title are satisfied via patching unicast channels at the LSA mini-server, the required unicast bandwidth during one time slot W_k is

$$\begin{aligned} BW_P &= \tau \cdot C + 2\tau \cdot C + \cdots + \beta\tau \cdot C \\ &= \frac{\beta(\beta+1)}{2}\tau \cdot C, \end{aligned} \tag{10}$$

where $\beta = \lfloor (W_k - \delta)\lambda_k \rfloor$ is the mean number of requests that arrive during one time slot and necessitate the usage of unicast channels. On the assumption of Poisson process, the probability density function of τ is

$$f(\tau) = \lambda_k \mathrm{e}^{-\tau \cdot \lambda_k}. \tag{11}$$

Hence, the average value of unicast bandwidth demand is

$$BW_P^{\text{avg}} = \int_0^{W_k - \delta} \frac{\beta(\beta+1)}{2} \cdot C \cdot \tau f(\tau)\mathrm{d}\tau. \tag{12}$$

In case of the proposed system, two cases are envisioned: $(0 \leqslant \tau \leqslant \theta)$ and $(\theta < \tau \leqslant W_k - \delta)$. In the former case, the first $\zeta = \lfloor \theta\lambda_k \rfloor$ requests will be assigned unicast channels, while the remaining $\beta - \zeta$ requests will be satisfied from the buffers of the previous users. The required unicast bandwidth in case of $0 \leqslant \tau \leqslant \theta$ is thus

$$BW_S = \frac{\zeta(\zeta+1)}{2}\tau \cdot C. \tag{13}$$

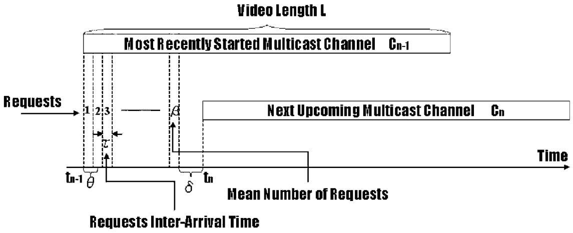
Video Length L
Most Recently Started Multicast Channel C_{n-1}
Requests
1
2
3
β
Next Upcoming Multicast Channel C_n
τ
Time
t_{n-1}
θ
δ
t_n
Mean Number of Requests
Requests Inter-Arrival Time

Fig. 9. Request inter-arrival times.

In case of $\theta < \tau \leqslant (W_k - \delta)$, the first request will be assigned a unicast channel to transmit τ time units' worth of data, while the upcoming requests will be satisfied from their neighbors' buffering. Hence, the required unicast bandwidth is

$$BW_S = \begin{cases} \frac{\zeta(\zeta+1)}{2}\tau \cdot C, & 0 \leqslant \tau \leqslant \theta, \\ \tau \cdot C, & \theta < \tau \leqslant (W_k - \delta). \end{cases} \tag{14}$$

On the assumption of Poisson process, the average of unicast bandwidth requirements in case fixed nodes are serviced via the neighbors-buffering policy is

$$\begin{aligned} BW_S^{\text{avg}} &= \int_0^{\theta} \frac{\zeta(\zeta+1)}{2} C \cdot \tau f(\tau)\mathrm{d}\tau \\ &\quad + \int_{\theta}^{W_k - \delta} C \cdot \tau f(\tau)\mathrm{d}\tau. \end{aligned} \tag{15}$$

The normalized disk-bandwidth reduction factor, Γ, of each session is plotted in Figs. 10 and 11. The numerical results show clearly that significant reductions in the disk bandwidth can be achieved in case of popular video items. Recall that the N_v sessions are classified according to the viewing popularity of their video titles. Fig. 10 illustrates the variation of the bandwidth reduction factor for different arrival rates. The number of multicast channels, N_m, is fixed to 300. The figure indicates that as the number of requests increases, so does the reduction factor for all the considered sessions. To investigate the impact of multicast channels on the system performance, we plot the reduction factor for different numbers of multicast channels in Fig. 11. The arrival rate is fixed to 1.5. It is observed that the reduction factor decreases for larger values of multicast channels. This decrease can be explained in terms of the length of time slots, $\{W_k,\ k = 1,2,\ldots,N_v\}$, of each session. Indeed, larger values of the total number of multicast channels lead to higher numbers of multicast channels assigned for each session and thus shorter time slots. Consequently, the number of requests to be satisfied via the neighbors-buffering policy, during a single time slot, decreases. This ultimately results in a decrease in the reduction factor. The two figures show also that the reduction factors of the two first popular sessions are slightly smaller than the reduction factors of the third and fourth

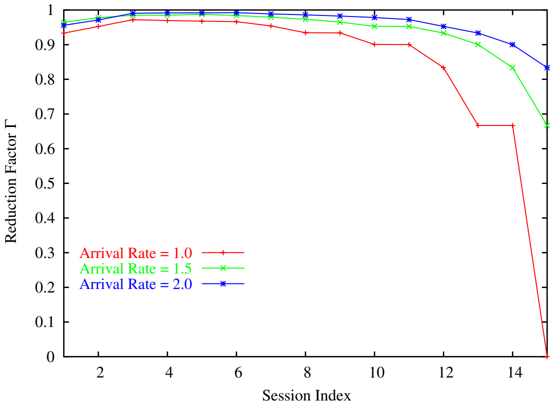

Fig. 10. Reduction factor (Γ) vs. session index: multicast channels number $N_m = 300$.

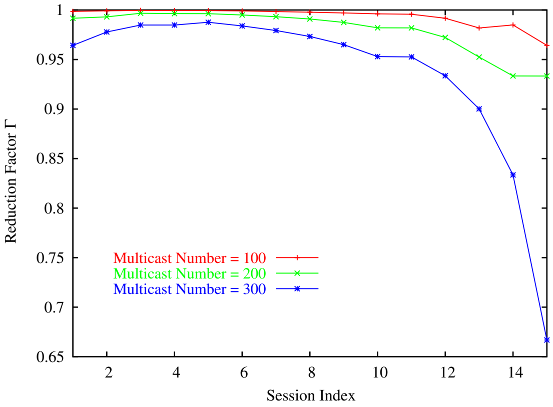

Fig. 11. Reduction factor (Γ) vs. session index: arrival rate $\lambda = 1.5$.

sessions. This performance is mainly due to the neighbors-buffering admission thresholds, δ and θ. In fact, requests to be satisfied via neighbors-buffering policy are those that come during the time interval [θ: $W_k - \delta$]. In case of the two first popular sessions, this time interval is significantly short and requests to be satisfied via neighbors buffering is accordingly small. This results in a relatively smaller values of reduction factor in comparison with the other consecutive popular sessions, as is indicated in the figure.

7.2. Numerical results: gain in LSA mini-servers capacity

As has been explained earlier, the service times of requests arriving at the LSA mini-server depend on their arrival time t_r and the start time t_{n-1} of the most recently started multicast channel C_{n-1}. Since $0 \leqslant t_r - t_{n-1} \leqslant W_k - \delta$, the service times for requests entering the unicast channel queue can be assumed uniformly distributed over the time interval [0: $W_k - \delta$]. The N_u unicast channels can be thus modeled as a M/M/N/$N + n$ queue [43], where N is the queue capacity ($N = N_u$). No queuing is assumed in the analysis ($n = 0$), for the simple reason that queuing may cause longer service response delays in case of high arrival rates, which may ultimately effect the short-latency nature of VoD service [12]. We compute numerical results from the M/M/n/$n + N$ queuing model to evaluate the capacity of the LSA mini-servers in the proposed architecture in terms of blocking probability.

To better estimate the system performance under heavier loads, the arrival rate is fixed to 2.0. Fig. 12 illustrates the blocking probability for different number of unicast channels. The figure indicates that the blocking probability decreases as more unicast channels are available at the mini-server. In case of satisfying users requests only via patching unicasts, the blocking probability is always in the vicinity of one and the system capacity is accordingly limited for all the considered unicast channels numbers. In case of the proposed

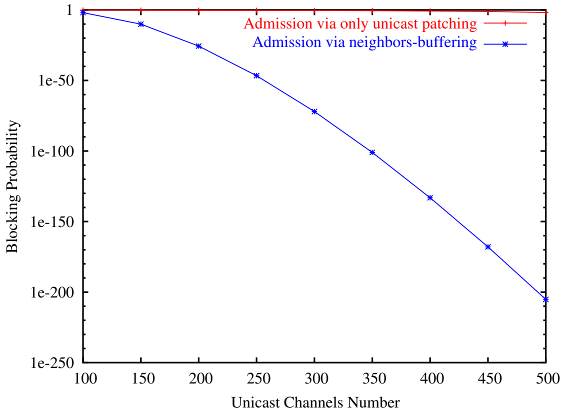

Fig. 12. Blocking probability vs. unicast channels number N_u (arrival rate $\lambda = 2.0$).

system, the blocking probability maintains always small values. Admittedly, when only a small number of unicast channels are available at the mini-server, the system capacity is limited. However, in case of large numbers of unicast channels, the proposed system experiences significantly low values of blocking probabilities and has thus the potential of satisfying a significant number of users.

7.3. Numerical results: control traffic for gathering users' status

The importance of the proposed system in terms of reducing the server bandwidth requirement and increasing the system capacity has been verified by numerical results. The price that the system pays for these advantages is the generated control traffic. This concern can be, however, addressed by carefully choosing an appropriate number of multicast channels at the MSA server and the user density at LSA areas.

In admission via neighbors-buffering policy, when the *i*th new user desires to join a particular session group, the user firstly issues a request packet to the service manager. In response, the service manager sends a control packet to the user including information on the session's multicast address and its size. The user will then multicast a single *session packet* to all the members of the session. In response, each member sends back a reply packet giving rise to $(i-1)$ *reply packets*. In the end, the user reports the one-way distance information to the service manager via a *report packet*. The total number of control packets generated upon the arrival of the *i*th new request is thus

$$CP_i = i + 3. \tag{16}$$

On the assumption of Poisson process, the total number of control packets generated for joining the session of the *k*th video title during one time slot W_k is

$$CP = \sum_{j=1}^{\alpha} CP_j = \frac{\alpha(\alpha+7)}{2}, \tag{17}$$

where $\alpha = \lfloor (W_k - \delta - \theta)\lambda_k \rfloor$ is the mean number of requests that arrive during one time slot. Using the probability density function of τ, the average rate of control packets generated for each session is

$$R_{CP}^{\mathrm{avg}} = \frac{1}{(W_k - \delta - \theta)} \int_0^{W_k-\delta} \frac{\alpha(\alpha+7)}{2} \cdot \lambda_k \mathrm{e}^{-\tau\lambda_k} \mathrm{d}\tau. \tag{18}$$

First is an investigation of the impact of the arrival rate on the average rate of control packets. Fig. 13 plots the average rate of control packets of the N_v sessions for different arrival rates. The number of multicast channels is fixed to 300. The figure demonstrates an obvious observation: higher arrival rates generate higher amounts of control packet traffic. To investigate the impact of multicast channels on the generated traffic, the average rate of control packets is plotted for different number of multicast channels in Fig. 14. Considering the case of heavy loads, the arrival rate is fixed to 2.0. The figure shows that allocating too few multicast channels leads to larger values of time slots and consequently larger amounts of control traffic. Whereas, allocating too many multicast channels reduces significantly the generated control traffic. Increasing the number of multicast channels would, on the other hand, trade off the obtained gain in terms of server bandwidth reduction, as is discussed earlier. Interestingly, the two figures indicate that the generated traffic increases according to the session popularity till it reaches its peak at the fifth session. It, henceforth, starts decreasing. Similar to the reduction factor, the reason behind this result can be explained in terms of the two admission thresholds, δ and θ. For highly popular video titles, the time interval $[\theta\colon W_k - \delta]$ is significantly short and the control packet traffic generated during this period of time is consequently small. For less popular items, requests arrival rate (λ_k) is small and the generated traffic rate is thus minimal.

On the other hand, in the proposed system, the service manager attempts always to satisfy a new user by establishing an *inter-connection* between the user and the nearest *neighbor* to transmit one portion of the video data while the rest is delivered to the user through a multicast channel directly from the satellite system. If a large number of requests are satisfied in a similar way, then reduction in the backbone LSA bandwidth requirement can

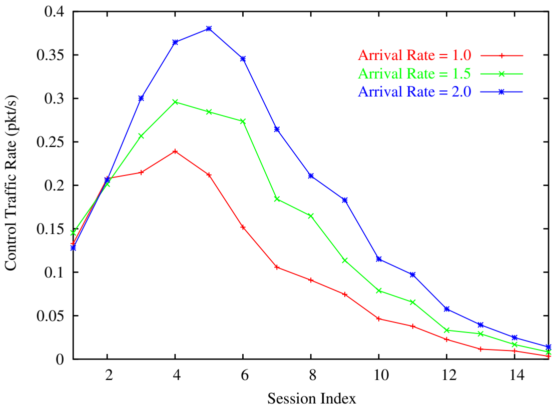

Fig. 13. Control traffic rate vs. session index: Multicast channels number $N_m = 300$.

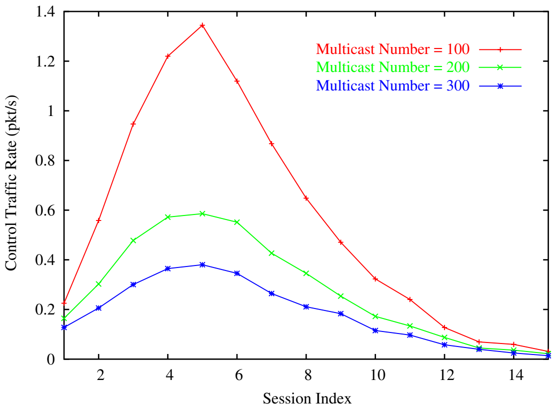

Fig. 14. Control traffic rate vs. session index: Arrival rate $\lambda = 2.0$.

be significant. Note that this reduction in the bandwidth requirement yields reduction in the video traffic, which is inherently bursty over both short and long time scales, and usually consumes a large amount of transmission bandwidth even after compression. This reduction yields also an alleviation of the congestion in both the network and the bottleneck due to the local video server. Reasoning so, the generated control traffic becomes a small cost paid to achieve these advantages.

7.4. Efficiency of the proposed system in handling handoffs

The remainder of this section concerns mobile nodes and verifies how the proposed system is efficient in smoothly handling handoffs. The performance evaluation relies on computer simulation, using Network Simulator (NS) [44]. Therefore, particular attention is paid to the design of an accurate and realistic simulation setup, which is described below, justifying the choices made along the way.

For the sake of simplicity, each LSA network is modeled as a single cellular network of a fixed size, and a MSA network is assumed as the set of these cellular networks. The coverage radius of each LSA is set to 10 miles. The link delay between the mini-servers of two adjacent LSAs is set to 30 ms. This experiment considers the case of a video title multicasted over a number of staggered multicast channels offset by a time slot worth $W = 10$ min. The conducted simulations consist of N mobile nodes roaming within a particular LSA. All the N mobile nodes are assumed to issue requests to join the session within a single time slot. The requests inter-arrival time is assumed to follow an exponential distribution with an average of 500 ms. The initial location of mobile nodes upon issuing a request is chosen randomly within the LSA area. The N mobile nodes are assumed to perform handoff between two adjacent LSAs at different times. The moving speed of mobile nodes is, thus, deliberately derived from a uniform distribution. In NS implementation, the minimum and maximum values of the distribution are set to a slow node moving speed, 5 mph, and a high node moving speed, 70 mph, respectively (*max_* = 70 mph, *min_* = 5 mph). Without loss of generality, it is assumed that handoffs occur between two neighboring LSAs (Inter-LSA handoffs) since it is the most common case.

The video traffic is generated by sending 4 UDP packets every 33 ms. The size of video packets is fixed to 1024 bytes. Bandwidth of unicast channels at all LSAs are assumed to be identical. At each LSA mini-server, the number of unicast channels is assumed to be largely sufficient to satisfy requests from all mobile nodes. On every handoff of a mobile host, statistics such as handoff occurrence time, T_h, and reception time of the first

Table 1
Simulation parameters

Factor	Simulation parameters and range of values
LSA coverage radius	10 miles
Delay between LSA servers	30 ms
Slot time	10 min
Mobile speed	5–70 mph
Mobile nodes	50–750
Video traffic	4 UDP pkts/33 ms
Packet size	1024 bytes

packet from the new LSA mini-server, T_r, are collected. In the simulation, the handoff occurrence time is the time when the mobile node switches its CoA registration when it passes the middle line of the overlapping area between two adjacent LSAs. All results are an average of seven simulation runs. Table 1 shows a complete list of the simulation parameters and the range of values studied.

A handoff is considered to be successfully handled if the time elapsed since the handoff occurrence time till the time when the mobile host starts receiving data from the new LSA mini-server is less than the playout delay, Δ. In other words, a handoff handling operation is considered a success, if

$$T_r - T_h \leqslant \Delta. \tag{19}$$

Note that the time ($T_r - T_h$) includes the network delay, waiting time due to prefetch buffering, location update time, and time required by a mobile host to recover from a packet loss in case of loss occurrence. To minimize the effects of packet drops on the system performance, numerous experiments conducted in [39,38] have recommended the setting of the playout delay to values larger than 100 ms. To demonstrate the efficiency of the system in smoothening handoffs, the following parameter is defined

$$\Phi = \frac{N_s}{N_h} \cdot 100, \tag{20}$$

where N_h and N_s denote the total number of mobile nodes that performed handoff during the time slot W and the number of successful handoff handling operations, respectively.

As is explained earlier, upon an Inter-LSA handoff occurrence, the old LSA service manager acknowledges the new service manager in the neighboring LSA, of the range of frames the mobile node may need. The prediction of this range of necessary frames may be inaccurate and results in the transmission of duplicate packets. Fig. 15 depicts a simple example of duplicate packets occurrence (e.g., packets C & D). To evaluate the system efficiency in terms of the number of duplicate packets, we define the transmission efficiency of the system, Ψ, as the ratio of the number of no redundant packets to the total number of transmitted packets averaged over the number of handoffs, N_h. For each mobile node k, $\{k = 1, 2, \ldots, N_h\}$, let N^k_{total} and $N^k_{\text{duplicate}}$ denote the total number of packets received by the mobile node k and the total number of duplicate packets computed over the time interval [T_h: T_h+200 ms], respectively. The system efficiency, Ψ, is expressed as

$$\Psi = \frac{1}{N_h} \cdot \sum_{k=1}^{N_h} \left(1 - \frac{N^k_{\text{duplicate}}}{N^k_{\text{total}}}\right) \cdot 100. \tag{21}$$

The percentage of successes, Φ, is plotted as a function of the total number of mobile nodes N in Fig. 16. The figure indicates that higher values of the playout delay absorb all transient effects during handoff and are able to smoothen further the handoff handling operation. A playout delay

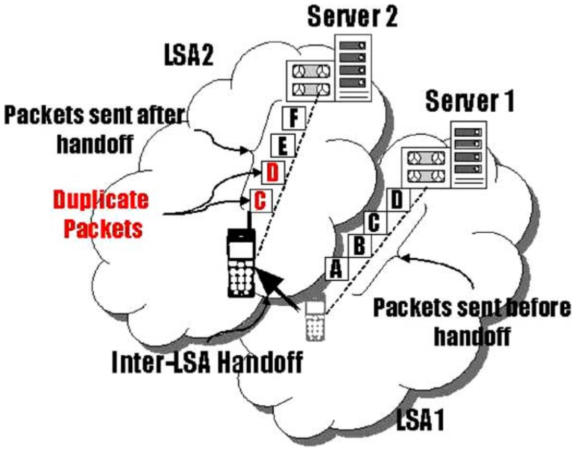

Fig. 15. A scenario of duplicate packets occurrence.

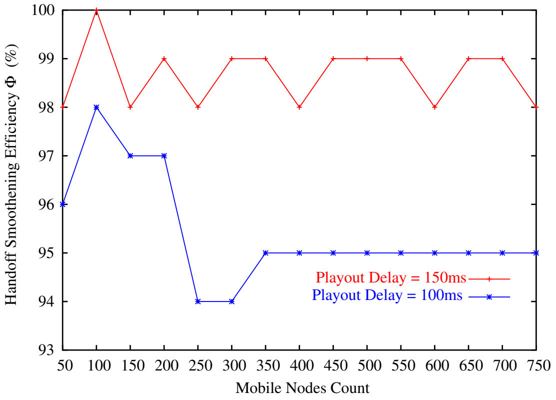

Fig. 16. Handoff smoothening efficiency Φ vs. mobile nodes count.

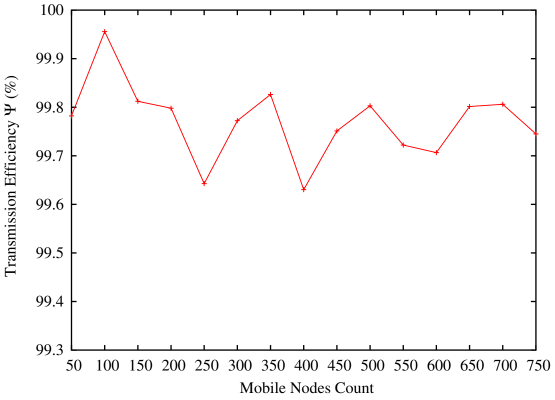

Fig. 17. Transmission efficiency Ψ vs. mobile nodes count.

of few seconds is seen sufficiently reasonable and tolerable because of the relatively long video length. Fig. 17 plots the transmission efficiency vs. the total number of mobile nodes. For all conducted simulations, the figure demonstrates the efficiency of the system in avoiding duplicate packets as the transmission efficiency, Ψ, remains always in the vicinity of 100%. The transmission efficiency and number of successes results show a little resemblance and inspire thus a certain correlation between the two measures. Indeed, small values of Φ mean that a certain number of mobile nodes had to wait for a longer period of time untill transmission becomes available from the new server. This longer wait-time may be due to delay in reporting handoff, recovery from packet losses, or queuing. During this period of time, the old server keeps on sending packets that the new server may be requested to transmit again. This would lead to higher number of duplicate packets and would be consequently translated into small values of transmission efficiency, Ψ.

8. Conclusion

In this paper, we have proposed an architecture based on a combination of Quasi-GSO satellite systems and existing terrestrial networks for building a very large-scale and efficient VoD system. A hierarchical distributed architecture has been considered. The coverage area of a Quasi-GSO satellite system is divided into a number of Metropolitan Service Areas, each comprising a single metropolitan VoD server. Popular video titles are stored at the MSA server and repeatedly transmitted on staggered multicast channels. In turn, each MSA is subdivided into a number of Local Service Areas according to geographical proximity and user density. To enable the delivery of digital media to a wide audience and to enhance the consumer experience, the LSA clusters can be thought of as IP transport networks as well. Each LSA contains a service manager and a mini-server. Only initial portions of video items are replicated at LSA mini-servers.

Three types of end-users have been considered, namely individual users in remote areas outside the reach of terrestrial networks, fixed nodes within terrestrial networks, and mobile nodes. For VoD services delivery, a set of request-admission techniques has been introduced for each end-user type. For individual users in remote areas beyond the reach of terrestrial networks, requests are simply satisfied via the Multicast Policy. In case of fixed nodes within the reach of terrestrial networks, if a request comes in between staggered start times of two adjacent multicast channels, the user joins the most recently started multicast session and then requests the missing part from a nearby neighbor instead of getting it from a patching unicast channel at the LSA mini server. This type of user must have enough buffer space to buffer data between staggered transmissions. The impact of this policy is to reduce the effective request arrival rate to the LSA mini-server. Savings in unicast channels at the LSA mini-server are exploited to satisfy requests coming from mobile nodes. In case of mobile nodes, to allow users to receive their VoD applications with a higher degree of mobility, handoff related issues are discussed and a simple scheme is proposed to guarantee a smooth streaming of video data when mobile users perform handoffs.

The importance of the proposed architecture is verified by numerical results. In case of requests coming from fixed nodes within the reach of terrestrial networks, analytical results elucidate the good performance of the architecture in terms of increasing the system scalability and reducing the disk-bandwidth requirements. Conducted simulations indicate also how efficient the proposed system is in handling handoffs. Overhead of the proposed system in terms of the generated control traffic is also evaluated. It is concluded that this cost can be addressed by carefully sizing LSA areas and choosing an appropriate number of multicast channels at the MSA server. The cost can also be justified as a reduction in the backbone LSA network load and resource savings increase dramatically at higher arrival rates.

It should be emphasized that there are several implementation issues that must be resolved when applying the proposed system in practice. For instance, it is assumed that fixed nodes within the reach of terrestrial networks have sufficient bandwidth resources to stream video data among themselves. This assumption is not valid for current networks where upstream bandwidth is limited (cable modem or ADSL subscribers). We, however, believe that with the rapidly on-going advances in bandwidth and transmission technologies, this limitation will be overcome in the near future. In addition to investigating the possibility of using the proposed system in delivering media of short durations (e.g., video clips) to mobile nodes, the authors are currently working on a prototype of the proposed architecture and its implementation in practice.

References

[1] P. Chitre, F. Yegenoglu, Next-generation satellite networks: architectures and implementations, IEEE Commun. Mag. 37 (3) (1999) 30–36.

[2] F. Carducci, G. Losquadro, The EuroSkyWay worldwide system providing broadband service to fixed and mobile end-users, Int. J. Satell. Commun. 17 (1999) 143–154.

[3] M. Holzbock, Y.F. Hu, A. Jahn, M. Werner, Evolution of aeronautical communications for personal and multimedia services, IEEE Commun. Mag. 41 (7) (2003) 36–43.

[4] T. Le-Ngoc, V. Leung, P. Takats, P. Garland, Interactive multimedia satellite access communications, IEEE Commun. Mag. 41 (7) (2003) 78–85.

[5] I. Minei, R. Cohen, High-speed internet access through unidirectional geostationary satellite channels, IEEE J. Select. Areas Commun. 17 (2) (1999) 345–359.

[6] H.D. Clausen, H. Linder, B. Collini-Nocker, Internet over direct broadcast satellites, IEEE Commun. Mag. 37 (6) (1999) 146–151.

[7] S. Yoshida, H. Kimura, Y. Inoue, T. Masamura N. Yamauchi, Interactive multimedia communication systems for next-generation education using asymmetrical satellite and terrestrial networks, IEEE Commun. Mag. 37 (3) (1999) 102–106.

[8] F.J. Ruiz, A. Fernandez, C. Miguel, L. Vidaller, A. Martinez, J.A. Carral, Multimedia systems based on satellite technology, Computer Networks ISDN Syst. 30 (16–18) (1998) 1.

[9] S. Dogan, A.H. Sadka, A.M. Kondoz, Video transmission over mobile satellite systems, Int. J. Satell. Commun. 18 (2000) 185–205.

[10] A. Jamalipour, The Wireless Mobile Internet: Architectures, Protocols and Services, Wiley, New York, 2003.

[11] M. Tanaka, K. Kimura, S. Kawase, H. Wakana, Applications of the Figure-8 satellite system, Space Commun. 16 (2000) 215–226.

[12] V.O.K. Li, Performance model of interactive video-on-demand systems, IEEE J. Select. Areas Commun. 14 (6) (1996) 1.

[13] C.C. Aggarwal, J.L. Wolf, The maximum factor queue length batching scheme for video-on-demand systems, IEEE Trans. Comput. 50 (2) (2001) 1.

[14] C.C. Aggarwal, J.L.Wolf, P.S. Yu, On optimal batching policies for video-on-demand storage servers, in: Proc. Int. Conf. Multimedia Systems, June 1996, pp. 253–258.

[15] S. Sheu, K.A. Hua, Virtual batching: a new scheduling technique for video-on-demand servers, in: Proc. 5th Int. Conf. Database Systems for Advanced Applications, Melbourne, Australia, April 1997, pp. 481–490.

[16] S. Sheu, K.A. Hua, W. Tavanapong, Chaining: a generalized batching technique for video-on-demand systems, in: Proc. Multimedia Computing and Systems, Ottawa, ON, Canada, 3–6 June 1997, pp. 110–117.

[17] L.S. Juhn, L.M. Tseng, Harmonic broadcasting for video-on-demand service, IEEE Trans. Broadcasting 43 (3) (1997) 268–271.

[18] Y. Birk, R. Mondri, Tailored transmissions for efficient near-video-on-demand service, in: Proc. IEEE Int. Conf. Multimedia Media Computing and Systems, Florence, Italy, June 1999.

[19] S. Viswanathan, T. Imielinski, Metropolitan area video-on-demand service using pyramid broadcasting, ACM Multimedia Syst. 4 (4) (1996) 197–208.

[20] K. Hua, Y. Cai, Patching: a multicast technique for true video-on-demand services, in: Proc. ACM Multimedia, September 1998.

[21] Y. Cai, K. Hua, K. Vu, Optimizing patching performance, in: Proc. SPIE/ACM Conf. Multimedia Computing and Networking, San Jose, CA, January 1999, pp. 204–215.

[22] S.W. Lau, J.C.S. Lui, L. Golubchik, Merging video streams in a multimedia storage server: complexity and heuristics, Multimedia Syst. 6 (1) (1998) 29–42.

[23] L. Golubchik, J.C.S. Lui, R.R. Muntz, Adaptive piggybacking: a novel technique for data sharing in video-on-demand storage servers, ACM Multimedia Syst. 4 (30) (1996) 14–55.

[24] J.Y.B. Lee, UVoD: an unified architecture for video-on-demand services, IEEE Commun. Lett. 3 (9) (1999) 277–279.

[25] J.Y.B. Lee, On a unified architecture for video-on-demand services, IEEE Trans. Multimedia 4 (1) (2002) 38–47.

[26] J.Y.B. Lee, C.H. Lee, Design, performance analysis, and implementation of a super-scalar video-on-demand system, IEEE Trans. Circ. Syst. Video Technol. 12 (11) (2002) 1.

[27] T. Taleb, N. Kato, Y. Nemoto, Neighbors-buffering based video-on-demand architecture, Signal Processing: Image Commun. 18 (7) (2003) 515–526.

[28] T. Taleb, N. Kato, Y. Nemoto, A neighbors-buffering based technique to provide scalable distance learning service in a multicast environment, in: Proc. of the 4th Int. Conf. Information Technology Based Higher Education and Training, July 2003.

[29] T. Little, D. Venkatesh, Prospects for interactive video-on-demand, IEEE Multimedia 1 (3) (1994) 14–25.

[30] C. Griwodz, M. Bar, L.C. Wolf, Long-term movie popularity models in video-on-demand systems, in: Proc. 1997 ACM SIGMM, Seattle, WA, November 1997, pp. 349–357.

[31] J.Y.B. Lee, Optimizing channel allocation in a unified video-on-demand system, IEEE Trans. Circ. Syst. Video Technol. 12 (10) (2002) 921–933.

[32] K.C. Almeroth, M.H. Ammar, The use of multicast delivery to provide a scalable and interactive video-on-demand service, IEEE J. Select. Areas Commun. 14 (6) (1996) 1.

[33] W. Liao, V.O.K. Li, The split and merge protocol for interactive video-on-demand, IEEE Multimedia 4 (4) (1997) 51–62.

[34] C.K. Miller, Multicast Networking and Application, Addison-Wesley, Reading, MA, 1999.

[35] D.L. Mills, Network Time Protocol (version 3), RFC 1305, March 1992.

[36] A.T. Campbell, J. Comez, S. Kim, Turanyi, C.Y. Wan, A. Valko, Comparison of IP micro-mobility protocols, IEEE Wireless Commun. 9 (1) (2002) 72–82.

[37] R. Koodli, Fast Handovers for Mobile IPv6 (Work in Progress), Internet Draft, Internet Engineering Task Force, 2002.

[38] R. Ramjee, K. Varadhan, L. Salgarelli, S. Thuel, S.Y. Wang, T. La Porta, HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks, IEEE/ACM Trans. Networking 10 (3) (2002) 396–410.

[39] Y. Pan, M. Lee, J.B. Kim, T. Suda, An end-to-end multipath smooth handoff scheme for stream media, in: Proc. of the 1st ACM Int. Workshop on Wireless Mobile Applications and Services on WLAN Hotspots, San Diego, CA, USA, 2003, pp. 64–74.

[40] A. Helmy, A multicast-based protocol for IP mobility support, in: ACM SIGCOMM 2nd Int. Workshop on Networked Group Communication, November 2000.

[41] A.T. Campbell, J. Gomez, S. Kim, Z. Turanyi, C.Y. Wan, A. Valko, Design, implementation and evaluation of cellular IP, IEEE Personal Commun. 7 (4) (2000) 42–49.

[42] R. Vadali, J. Li, Y. Wu, G. Cao, Agent-based route optimization for mobile IP, in: Proc. of IEEE VTC, October 2001.

[43] G. Bolch, Queueing Networks and Markov Chains, Wiley-Interscience, New York, 1998.

[44] UCB/LBNL/VINT, Network Simulator—ns (version 2). Available from <http://www.isi.edu/nsnam/ns/>.

Tarik Taleb is currently a Ph.D candidate at the Graduate School of Information Sciences, Tohoku University. He received his B.E. and M.E. degrees in computer science from Tohoku University in 2001 and 2003, respectively. His research interests lie in the field of wireless networking, especially in the performance of TCP over broadband satellite networks and satellite network security. His recent work has focused on multimedia transmission in multicast environments and reliable multicast protocols.

Nei Kato received his M.S. and Ph.D. degrees from the Graduate School of Information Sciences, Tohoku University in 1988 and 1991, respectively. He joined the Computer Center, Tohoku University in 1991 and now he is a professor at the Graduate School of Information Sciences, Tohoku University. He has been engaged in research on computer networking, wireless mobile communications, image processing and neural networks. He is a member of IEEE and the Information Processing Society of Japan.

Yoshiaki Nemoto received his B.E., M.E. and Ph.D degrees from Tohoku University, Sendai, Japan, in 1968, 1970 and 1973, respectively. Now he is a professor with the Graduate School of Information Sciences, and director of the Information Synergy Center, Tohoku University. He has been engaged in research work on microwave networks, communication systems, computer network systems, image processing and handwritten character recognition. He was a co-recipient of the 1982 Microwave Prize from the IEEE Microwave Theory and Techniques Society. He is a member of the IEEE and the Information Processing Society of Japan.