ARTICLE IN PRESS

Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Journal of Network and Computer Applications ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Journal of NETWORK and COMPUTER APPLICATIONS

www.elsevier.com/locate/jnca

QoS-enabled multicast for delivering live events in a Digital Cinema scenario

D. Di Sorte*, M. Femminella, G. Reali

Dipartimento di Ingegneria Elettronica e dell'Informazione (DIEI), University of Perugia, Via G. Duranti 93, 06125 Perugia, Italy

Received 24 July 2007; received in revised form 9 January 2008; accepted 20 February 2008

Abstract

Digital Cinema (DC) consists of integration of new advanced digital technologies in the context of the cinema system. As regards the transport of DC content towards theatres, Distributors may select the method that is both economically and technically sound.

In this work, which is carried out within the framework of the IST Integrated Project Enhanced Digital CINEma (EDCINE), we deal with the network distribution service provided by a Network Service Provider, which becomes a new actor in the DC business. One of the main criticalities of the system is the very large size of the contents to be transferred towards theatres. From the operator's perspective, this criticality translates into the objective of optimising the usage of network resources while complying with quality of service (QoS) constraints.

The goal of this paper is to present the system which is able to support the network delivery of DC contents, with a special focus on live event delivery. This service can consume a large amount of network bandwidth, not only because of the volume of transmitted data, but also due to the number of receivers, and thus multicast transmission proves to be very useful. Consequently, a key issue of the overall distribution system is the request-routing algorithm, the goal of which is to optimise the QoS-guaranteed delivery of a number of live streams in the backbone, each one of which is sent towards a set of theatres (QoS multicast routing). We consider the MultiProtocol Label Switching mechanism, which has emerged as an elegant solution to meet traffic engineering and resource reservation requirements in backbone networks, and focus especially on the overall request-routing procedure, the mathematical modelling of the problem, and relevant solving algorithms. Finally, we

*Corresponding author. Tel.: +39 075 585 3630; fax: +39 075 585 3654.

E-mail addresses: disorte@diei.unipg.it (D. Di Sorte), femminella@diei.unipg.it (M. Femminella), reali@diei.unipg.it (G. Reali).

1084-8045/$ - see front matter
doi:10.1016/j.jnca.2008.02.004

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

present the comparative performance evaluation of these algorithms by means of an extensive simulation campaign performed with the OMNeT++ simulation platform.

Keywords: Digital Cinema; Network distribution architecture; Multicast; QoS; MPLS; Performance evaluation

1. Introduction

Digital Cinema (DC) consists of integration of new advanced technologies (such as high-performance film scanners, digital image compression algorithms, high-speed data networking and storage, and advanced digital projection) in the context of the cinema system; it includes the replacement of celluloid-based distribution and projection with digital technologies. A DC system can be divided into four stages: mastering, transport, storage and playback, and projection. At the mastering stage, the film is compressed, encrypted and packaged for delivery to theatres. The data is then transported to the exhibition site, where it is decrypted, uncompressed and played back Bilgin and Marcellin (2006).

Within this framework of technological achievements coming from industries and research organisations, Digital Cinema Initiatives LLC (DCI) Digital Cinema System Specification v1.1 (2007) is a new entity created by several studios, with the primary purpose of establishing uniform specifications for DC.

The DCI specifications should be transparent, published industrial standards, which are widely accepted and codified by national and international standards bodies. In particular, the Society of Motion Picture and Television Engineers (SMPTE) (SMPTE Web Site) and DCI selected JPEG 2000 (Information technology-JPEG, 2000) as the compression format for DC. In addition to this new compression scheme, the new Material eXchange Format (MXF) file format and security tools (encryption as well as the Digital Right Management system) can give rise to very large files. JPEG 2000 is less compression efficient than MPEG-2; however it uses a wider colour space, higher bit-depth and provides a bit stream with scalability in quality and resolution; one single, high-quality media could, therefore, be projected on a lower quality or resolution set-up without multiple encoding of the content.

A DCI film file is a systematic collection of compressed images, including some additional audio and data tracks; the uncompressed, original data represent the so-called Digital Cinema Distribution Master (DCDM). As far as only the video is concerned, its final compression rate corresponds to a total, encoded, maximum bit rate of 250 Mb/s (from the original rough bit rate around 1.5 Gb/s), whereas the audio is left uncompressed and can achieve a bit rate of approximately 20 Mb/s. After video compression, all tracks are encrypted and packaged into a particular format (see Fig. 1) called Digital Cinema Package (DCP), with a rate up to around 300 Mb/s.

As regards the transport of DC content, the DCI specifications Digital Cinema System Specification v1.1 (2007) state that the transport of DC content (defined as the movement of the packaged DC content) from the site of origin to the theatres can be accomplished in many different ways, and Distributors may select the method that is both economically and technically sound to ship their content to the theatres. This can include either the use of physical media or wired/wireless network distribution.

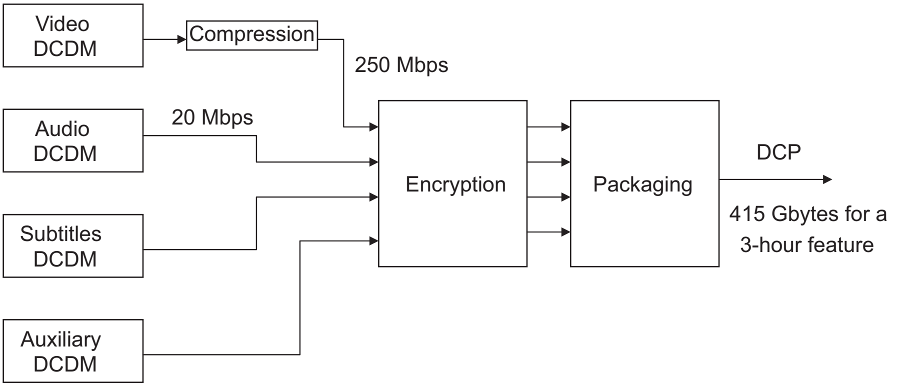

Fig. 1. DCI film package.

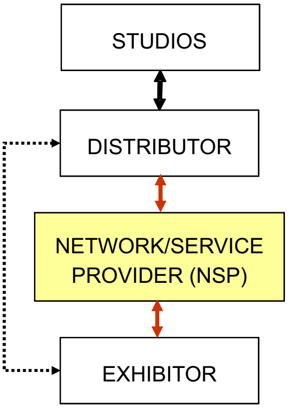

Fig. 2. DC distribution: contractual agreements.

In the latter case, Distributors rely on the service provided by a Network Service Provider (NSP), which, in this case, becomes a new actor in DC business (Fig. 2). In more detail, a Distributor (the intermediary between artistic content creators and artistic content Exhibitors) establishes agreements with one or more NSPs, which are in charge of storing and distributing DC contents. In turn, an Exhibitor must have an agreement with an NSP to be physically connected to the network.

Clearly, any selected distribution method must guarantee not only a secure environment for the content, but also no data corruption: the DCI system uses encryption in order to protect audiovisual contents from unauthorised copying and illegal distribution. In particular, the content owner's encryption is required to be irremovable during transport, and the files are required to retain all of the data of the original files. Furthermore, segmentation of the content may be carried out in order to accommodate contents in storage devices or to deal with bandwidth constraints.

Similar specifications have been delivered by European Digital Cinema Forum (EDCF) that recommends (EDCF, 2003) "… the distribution system has to be secure across any chosen transport mechanism. Transport can be via satellite, data tape, hard disk, fibre optic, etc."

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

So far, the service of a DC system is based on the download-and-play paradigm; the content is downloaded from a server in the network, then it is stored on a local server in the theatre, and finally it is projected.

DCI specifications have excluded the projection of live events, which, in our view, is an application that can bring new interest to theatre-going and thus stimulate the DC business. The issue of distributing live events (e.g., sports events, music events, exceptional events) for entertainment purposes has been overlooked by the standards committees and the IST Integrated Project Enhanced Digital CINEma (EDCINE) (IST EDCINE Project Web Site) aims to correct this by creating tools and proposing improvements to standards in order to tackle the issues.

As regards networking aspects, the data stream delivery to theatres has to be guaranteed in real time. The analysis of a survey submitted to European specialists from research institutes and companies in the field of DC (EDCINE Project Deliverable D1.1) showed that end-to-end delay should be below 1 s, packet loss rate has to be negligible (next to zero) and delay jitter should be in the order of tens of milliseconds. The stream bit rate has to be definitely lower than the bit rate of pre-recorded films (JPEG2000-based, with video rates from 80 to 250 Mb/s Digital Cinema System Specification v1.1, 2007). A higher compression ratio may lead to a value in the order of a few tens of Mbps, which can guarantee excellent video quality when using either MPEG-2 or MPEG-4 coding schemes. However, the high bit rates together with the hard quality of service (QoS) constraints call for Traffic Engineering (TE) and resource reservation mechanisms to enhance the best effort service provided by the Internet Protocol (IP). In this regard, MultiProtocol Label Switching (MPLS) Rosen et al. (2001) has emerged as an elegant solution to meet these requirements in IP backbone networks. Moreover, video streaming can consume a large amount of network bandwidth not only as a result of the volume of the transmitted data, but also due to the number of receivers; thus, multicast transmission is very useful in this scenario. In this framework, a key issue of the overall distribution system is the request-routing algorithm, the goal of which is to optimise the QoS-guaranteed delivery of a number of live streams, each one of which is sent towards a set of theatres (QoS-enabled multicast routing).

The goal of this paper is threefold:

- to design the system architecture able to support the network delivery of DC contents towards theatres, with a special focus on live events delivery;
- to analyse the request-routing problem for live events in the distribution network, with a special focus on the QoS-enabled multicast routing in the core network. In particular, we deal with the overall request-routing procedure, the mathematical modelling of the problem and relevant solving algorithms;
- to compare performance evaluation of the algorithms used to solve the multicast QoS routing optimisation problem. This task is accomplished by means of an extensive simulation campaign performed with a modified version of the OMNeT++ simulator (INET Framework for OMNeT++/OMNEST).

The paper is organised as follows. The next section reports some details about the network distribution system, describing the distribution network, the internal theatre network, and provides some insights into video coding rates. Section 3 describes the request-routing procedure, the mathematical model associated with it, and a number of different QoS-enabled multicast routing approaches. Section 4 presents a performance comparison among these approaches. Finally, Section 5 reports some concluding remarks.

2. Network distribution system

Within the framework of EDCINE, we have considered a Content Delivery Network (CDN) able to provide two kinds of network services: the distribution of pre-recorded films to theatres and the delivery of live events. The first is based on the download-and-play paradigm; the content (typically a film) is downloaded from the servers of the CDN, then it is stored on a local server, and finally it is projected. The QoS requirement is on the maximum download time. The second is relevant to the streaming of live events, and thus it is subject to hard requirements in terms of transfer and processing time. One of the main criticalities of the transport system is the huge size of the contents to be transported.

Clearly, Distributors may rely on the service provided by an NSP to manage the EDCINE CDN, which is an overlay network, whose elements are in charge of storing, distributing, delivering and routing capabilities. It consists of (Verma, 2002):

- a number of surrogate sites (SSs) storing contents, beyond the origin site (OS);
- a networking infrastructure between the CDN sites;
- a delivery infrastructure to move contents from sites to clients;
- a system able to route service requests;
- a policy to distribute and to manage contents among the OS and SSs;
- an accounting system to collect data to support procedures of statistical analysis, and billing.

Fig. 3 depicts the overall distribution system for both pre-recorded contents and live events. The internal links connecting the CDN systems are logical links, consisting of a number of physical links of the underlying network supporting the CDN service. Since these logical links can be seen as belonging to a private network (the CDN overlay) built on top of the underlying physical network, they are implemented via Virtual Private Networks (VPNs).

Within the EDCINE framework, encryption is a requirement for the transport of DC contents. Moreover, since the delivery of DC contents has to be QoS-guaranteed, it is necessary to provide VPN links with bandwidth guarantees. Thus, the VPNs in the EDCINE scenario will be "hybrid" (i.e., secure and QoS-guaranteed), according to the classification of VPNs defined by the VPN Consortium (The Virtual Private Network Consortium). Encryption is guaranteed at the application layer (i.e., security associations between mirrors and content servers within theatres have to be established off-line), whereas bandwidth provisioning in the core network can be provided through MPLS (Rosen et al., 2001). In fact, MPLS-based VPNs are emerging as the popular choice by service providers to build IP VPN due to their scalability, flexibility, cost and the ability to provide IP applications with QoS across the network (Rosen and Rekhter, 1999; Daniel, 2004). Within this framework, a Provider Edge (PE) node (an IP router or an MPLS-compliant router) is a device that connects customers to the provider's backbone network. The theatre site is a local private network (i.e., a LAN), the gateway of which (customer edge, CE) is connected to the PE via a high-speed access connection (e.g., VDSL2, fibre link, Metro Ethernet). In order to deliver the live events to multiple theatres

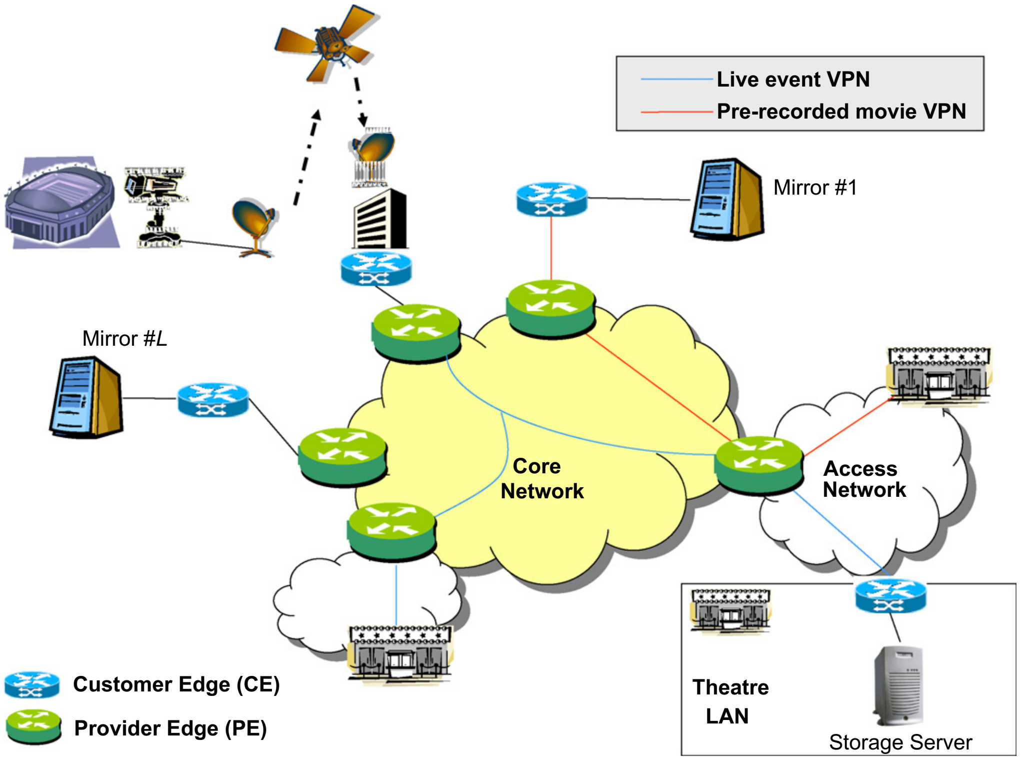

Fig. 3. Network distribution architecture.

contemporarily with the necessary quality, the backbone has to be able to provide multicast VPNs with QoS support (Morin, 2007). In this regard, MPLS is still a very interesting solution, since it has been recently updated with the capability of providing QoS-enabled, truly multicast-oriented paths Yasukawa (2006). More details on the core network, which is a key issue of this paper, are provided in Section 2.1.

Pre-recorded contents can be downloaded by Exhibitors from a set of mirrors, whereas a live stream enters the core network via the ingress router, to which the Head-End of the live event Distributor is connected, and reaches a number of theatres. Clearly, an NSP may have agreements with more than one live event Distributor (see Fig. 4), and thus the ingress points of the live streams can be different.

The request-routing system is a key element for the optimisation of the CDN services from both operator and user viewpoints.

As for pre-recorded film distribution, the routing system capability of a CDN (4–7 routing) is in charge of associating each service request from Exhibitors to a set of CDN mirrors. In other words, each request to download a film has to be routed towards one or more mirrors, according to a number of factors, such as service demand, mirror catalogue (content presence), server congestion, network resources and download time requirement.

As for live events delivery, the routing system capability of a CDN is in charge of mapping each stream associated with a live event with a point-to-multipoint VPN in the MPLS backbone. The multicast tree associated with a live event is clearly dependent on the

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

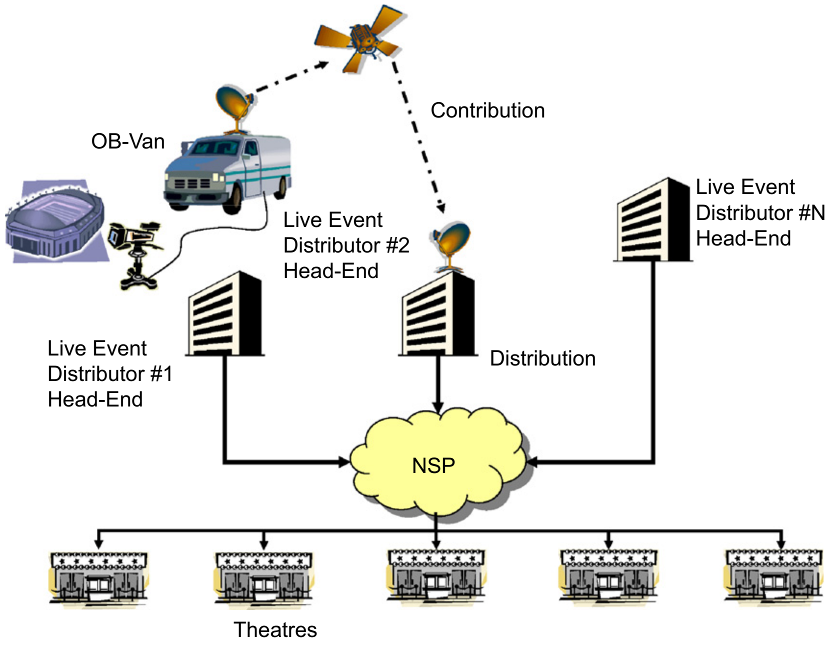

Fig. 4. Live event delivery system.

service requests from Exhibitors, the flow ingress point to the backbone and the relevant egress points (associated with the theatres requesting the service). In other words, each live stream has to be routed towards one or more theatres, accounting for service demand, network resource availability and QoS requirements.

In this paper, we especially deal with the multicast QoS routing of live streams within the core network. This issue is described in detail in Section 3.

2.1. Distribution network

MPLS is a protocol able to run below IP and on top of several layer 2 technologies (PPP, SDH/SONET, Ethernet). It enables connection-oriented paths (Label Switched Paths, LSPs) to be created within IP-based core networks. It de-couples routing from forwarding functions (Rosen et al., 2001). LSPs are set-up in advance by Label Edge Routers (LERs) at the domain ingress, and IP packets are classified when they enter the MPLS domain by adding short fixed-length labels. Packets belonging to the same Forwarding Equivalence Class are sent through the same LSP and Label Switched Routers (LSRs) process them according to the label only. Thus, MPLS enables fast packet switching and TE to be performed. TE is defined as those aspects dealing with the issue of performance optimisation of operational networks. TE is able to improve QoS by reducing delay and packet losses, while maintaining a high resource utilisation. TE also reduces the vulnerability of the network to possible service failures.

In order to satisfy the requirements for TE over MPLS, an extension of RSVP signalling protocol (RSVP-TE) has been defined (Awduche et al., 2001). RSVP-TE allows the dynamic establishment of LSPs, which can be automatically routed away from network failures,

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

congestion and bottlenecks. RSVP-TE also supports smooth re-routing of LSPs, preemption and loop detection. Furthermore, RSVP-TE is able to establish LSP tunnels with resource allocation along the path. An LSP can be set-up either with or without resource reservations. LSPs without resource reservations may be used to deliver best effort traffic, whereas LSPs with resource reservations can deliver real-time traffic with hard QoS.

Finally, MPLS-based VPNs are emerging as the popular choice by service providers to build IP VPN due to their scalability, flexibility, cost and the ability to provide IP applications with QoS across the network (The Virtual Private Network Consortium; Duffield et al., 1999; Daniel, 2004).

To sum up, MPLS is an effective solution to meet the requirements of IP backbone networks and allows resource optimisation and fast failure recovery. Even though it was limited to point-to-point LSPs, the need to support point-to-multipoint (P2MP) services using traffic-engineered LSPs has emerged. This requirement has motivated some enhancements of the basic MPLS-TE tools in order to support P2MP MPLS-TE LSPs (Yasukawa, 2006). A P2MP LSP is a unidirectional LSP (Awduche et al., 1999; Rosen et al., 2001) which has a single ingress LSR and one or more egress LSRs, and is unidirectional. An explicitly routed P2MP LSP consists of a number of paths, each from the ingress LSR to an egress LSR, defined as source-to-leaf sub-LSPs. They are set a priori (source routing), without requiring a multicast routing protocol in the backbone. IETF RFC 4875 Aggarwal et al. (2007) describes a solution to allow a non-ingress LSR to be a replication/branch LSR, able to replicate the incoming data on one or more outgoing interfaces. Thus, an explicitly routed P2MP LSP is set-up by grouping multiple source-to-leaf sub-LSPs and relying on data replication at branch nodes; such a solution uses RSVP-TE as signalling protocol, without requiring a multicast routing protocol in the core network. In more detail, each explicitly routed P2MP LSP should be set-up using the RSVP-TE Extension to P2MP LSP described in Aggarwal et al. (2007), which defines the P2MP_Secondary_Explicit_Route Object (SERO), in addition to the existing Explicit_Route Object (ERO) (Rosen et al., 2001), used to perform explicit unicast routing. The SERO is used by RSVP-TE to specify the explicit route of a source-to-leaf sub-LSP. Both ERO and SERO are carried in the RSVP-TE Path message.

As for the access section of the distribution network, we assume that theatres are connected to the NSP core network via a high-speed access connection, such as VDSL2, ADSL2+, fibre link, Metro Ethernet, able to support a continuous downstream with bit rates in the order of some tens of Mb/s.

2.2. Network inside theatres

The infrastructure deployed inside a multiplex theatre which exploits DC features (DC Theatre System, DCTS) commonly consists of a number of networked entities/devices connected by means of a high-speed switched LAN (Gigabit or 10 Gigabit Ethernet) (Digital Cinema System Specification v1.1, 2007). As mentioned above, the theatre is connected to the external network with a high-speed connection via an access device (CE).

The networked entities which make up a DCTS are as follows (see Fig. 5 and Digital Cinema System Specification v1.1, 2007):

- *Storage*: The content storage can be arranged in two basic configurations, central and local storage. The first foresees all the films to be stored in a central server, from which

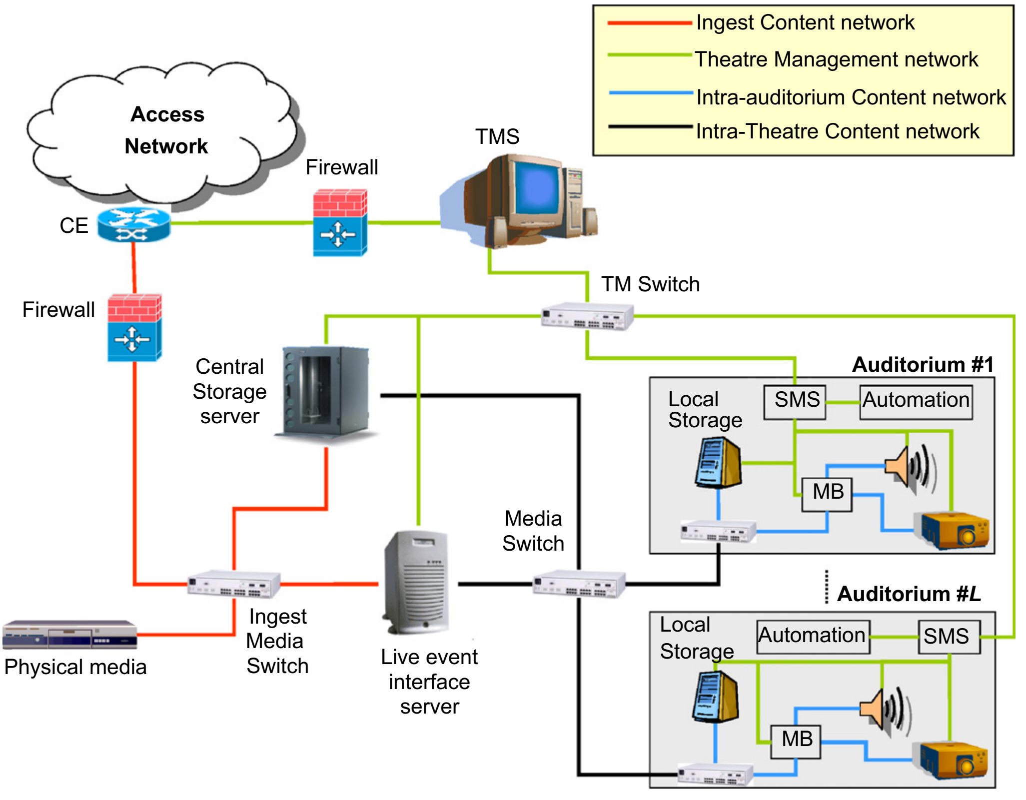

Fig. 5. Theatre LAN.

they are played back to the auditorium. The second implies a single storage system for each screen/auditorium. A combination of them may be deployed as well, as suggested in Digital Cinema System Specification v1.1 (2007).

- *Live event interface server*: It is the device in charge of receiving the raw, live data flow from the CE and de-multiplexing it into the live event streams. In addition, it has to deliver the streams to the auditoria. All these operations must be done in real time. The live event interface server is an additional device with respect to the configuration foreseen by the DCI specifications, which do not include live event projection.
- *Media block*: It is the entity responsible for converting the packaged, compressed and encrypted data into raw image, sound and subtitles. In addition, the media block is required to playback the image, audio and other time-dependent content in a manner that presents a synchronised performance to the audience. In the case of a live event projection, it will receive the contents directly from the live event interface server.
- *Projection system*: It is required to convert the uncompressed digital image data coming from the media block into the pictures to be projected to the audience on the screen.
- *Audio system*: It delivers the sound of the theatrical presentation to the audience. It is in charge of receiving the uncompressed digital audio from the media block, converting it into analogue and directing it to the speakers.

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

- *Screen Management System (SMS)*: It provides theatre management with a user interface for the local control of an auditorium. In addition, the SMS can monitor and run diagnostics on auditorium equipment and provide the network administrator with status information.
- *Theatre Management System (TMS)*: This is the central management system in charge of controlling all the SMSs and servers.
- *Automation*: The screen automation system interface with life safety, motor controlled curtains, and in general with all automated devices commonly installed in an auditorium.
- *Network infrastructure*: Inside the theatre, different LANs can be deployed to accomplish different tasks. Some of these local networks have to be (physically or logically) distinct for security reasons, to separate data transport from control/signalling messages. In more detail, the envisioned internal LANs are:
    - *Ingest content network*: This is the LAN via which the DCTS receives contents from the outside world. It enables films to be loaded on to the central storage server from either external networks (i.e., content loaded via the CE through a firewall) or physical removable media (hard disks or optical supports). In addition, it allows the live event interface server to receive contents from the access network via the CE.
    - *Theatre management network*: This LAN is in charge of transporting signalling and control/management messages from the TMS to the SMSs, central storage and live event interface server. In addition, it connects the SMS to all internal equipment inside each auditorium. Finally, such a LAN is also in charge of transporting the security keys. It has to be protected with a security firewall between the TMS and the CE.
    - *Intra-theatre content network*: This LAN is set aside to transporting DC contents from the central storage server or the live event interface server towards the auditorium LANs.
    - *Intra-auditorium content network*: This network is in charge of transporting contents inside the auditorium. In more detail, it enables pre-recorded contents to be loaded from the central storage server to the local storage before the show, and the contents to be played back from the local storage to the media block during the show. As for live events, the intra-auditorium content network enables the live stream coming from the live events interface server to be delivered directly to the media block. Finally, in both cases, this type of LAN is in charge of transporting decoded contents from the media block to audio and projection systems.

2.3. Video coding issues

Some additional words need to be spent on the video coding scheme.

The general framework foreseen for live events delivery is the one depicted in Fig. 4, where the live streams from different cameras are assembled by a local director in the mobile station (Outside Broadcast Van) just on site, and then delivered via satellite to the Distributor Headquarter. This is the so-called "contribution" part of the network. In the Headquarter, the data flow is re-elaborated (e.g., ciphered and/or further compressed) and then sent on a VPN from the CE of the Distributor Head-End to the CEs of the theatres involved. This is the so-called "distribution" part of the network.

There are two limitations to the bit rate of the live stream, one in the contribution network and the other in the distribution network. The first refers to the satellite transponder. In the DVB-S2 system, there is a bandwidth availability equal to approximately 55 Mb/s (Bertella et al., 2007), whereas in the DVB-S the bandwidth limit is approximately 30% lower. The second is due to the bandwidth limitations of the theatre access network. In the case of satellite access, we have the limits described above, whereas in the case of a wired connection, bandwidth availability depends on the specific technology deployed. For instance, in the case of VDSL2, the bandwidth quickly deteriorates from a theoretical maximum of 250 Mb/s at the source to 100 Mb/s at 0.5 km and 50 Mb/s at 1 km, and degrades at a much slower rate from there on. In the case of ADSL2+, the maximum speed is equal to 24 Mb/s.

Bearing in mind these bandwidth limitations, the JPEG2000 coding scheme (corresponding to rates from 80 to 250 Mb/s Digital Cinema System Specification v1.1, 2007) is unsuitable for live event network delivery. The MPEG-4 AVC Video Compression standard has recently been approved by the ITU and the ISO (also called MPEG4 Part10 or H.264, ITU-T Recommendation H.264, 2005). It enables bit-rate savings of the order of 50% with respect to MPEG-2 to deliver the same quality. If we consider the bandwidth limits in both contribution and distribution networks, the MPEG-4 profiles with 20 and 50 Mb/s (2K × 1K@30 fps) (ITU-T Recommendation H.264, 2005) are definitely good candidates to support DC live events.

3. Request-routing procedure for live events delivery

As the analysis of a survey submitted to European specialists from research institutes and companies in the field of DC (EDCINE Project Deliverable D1.1) has shown, the web interface has proved to be a suitable way to book live event delivery. This means that the NSP in charge of managing DC contents can provide a web server, via which each theatre administrator can request the access to a live event, which is typically characterised by a start hour and an end hour. In principle, more than one event can be scheduled at the same time.

As mentioned above, the routing system capability of the CDN has to map each stream associated with a live event with a P2MP LSP in the MPLS backbone, at the same time accounting for bandwidth availability and the QoS requirements needed to support a real-time transfer. In more detail, the problem of mapping a multicast tree into an explicitly routed P2MP LSP consists of ERO and SERO computation (see Section 2.1), starting from the output of the multicast allocation problem (QoS-enabled multicast routing), the mathematical model of which is presented in the following Section 3.1.

With reference to the sequence diagram depicted in Fig. 6, we list below the main steps of the procedure to manage the service demand coming from Exhibitors for a set of live events, the starting hours of which are the same:

- The Exhibitor books the live event in a predefined time window, clearly in advance with respect to the beginning of the event. For each multicast stream associated with a live event, the event start time, its time duration, the ingress point and egress points of the NSP backbone networks are known; none of them change during the event.
- The requests coming from the theatres are collected by an NSP server (decision maker, DM), which is also in charge of retrieving the availability of network resources from monitoring entities just before the request-routing algorithm execution.

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

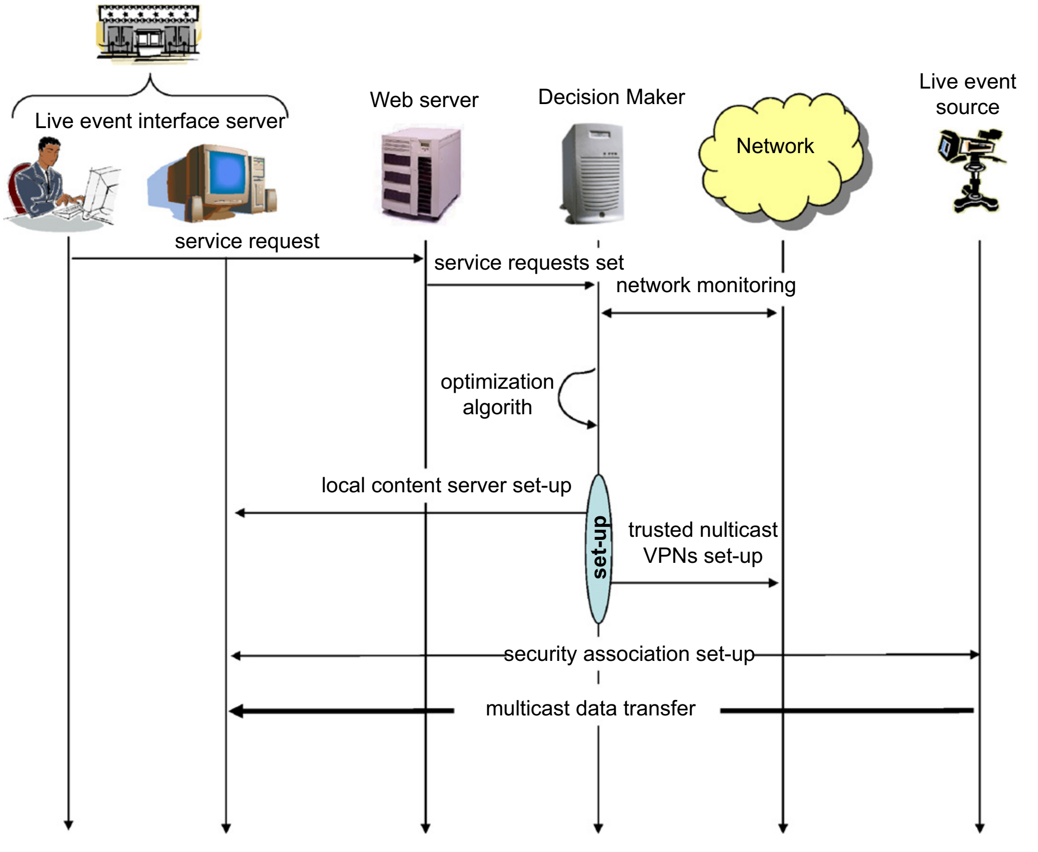

Fig. 6. Live events: sequence diagram.

- The DM runs the multicast QoS routing algorithm to compute the optimal resource allocation within the core network.
- The DM is in charge of setting up the multicast tree for each flow.

The overall procedure is repeated periodically, according to the live events schedule.

Below, in Section 3.1, we present the mathematical formulation of the QoS-enabled multicast routing problem and classify it according to the taxonomy of the multicast routing problems proposed in Wang and Hou (2000). Then, in Section 3.2, we discuss a number of approaches able to solve such a problem, and classify them again according to the taxonomy of the multicast routing algorithms as shown in Wang and Hou (2000).

3.1. Mathematical model of the QoS-enabled multicast routing problem

In this section, we present the mathematical formulation of our reference QoS-enabled multicast routing problem over a given network configuration. The goal of the NSP administrator is to optimise the QoS-guaranteed delivery of a number of live streams, each one of which is sent towards a set of theatres.

The considered network topology represents the core network, because it is possible to find alternative paths connecting sources with destinations only in the core network. In fact, all sources/sinks access the ingress/egress routers of the backbone through static, dedicated access links. The core network topology can be modelled as an oriented graph $G = (V,E)$, where V is the set of vertices with $|V| = n$ and E is the set of edges with $|E| = m$, where $e_k = (p,w)$ is the oriented arch from node p to node w. We assume the edges are numbered from 1 to m. Each multicast flow i is characterised by the triple (s_i, R_i, b_i), $i = 1, \ldots, L$, where s_i is the source node of the multicast flow, R_i is the set of receivers with $|R_i| = r_i$ and b_i is the amount of bandwidth required to transport the ith flow with the necessary QoS guarantees. We assume to reserve an amount of bandwidth equal to the peak rate.

The output of the multicast QoS optimisation problem is a set of multicast trees, each of which is associated with a live stream. Clearly, each multicast tree has to be converted into an explicitly routed bandwidth-guaranteed P2MP LSPs.

Note that the DC scenario is static and thus quite different from the QoS multicast problems which account for a dynamic join and prune of leaves (e.g., see Kodialam et al., 2003). This allows considering all streams concurrently in the problem formulation.

With reference to the previous definitions of oriented graph and flow, Table 1 reports the list of system parameters, classified into inputs and outputs of the problem.

The constraints of the problem are:

$$\sum_{i=1}^{L} Y_{ki} b_i \leqslant B_k, \quad \forall k = 1, \ldots, m, \tag{1}$$

$$X^i_{kp} \leqslant Y_{ki}, \quad \forall i = 1, \ldots, L, \quad \forall k = 1, \ldots, m, \quad \forall p = 1, \ldots, r_i, \tag{2}$$

$$MX^i = A^i, \quad \forall i = 1, \ldots, L, \tag{3}$$

$$X^i_{kp}, Y_{ki} \in \{0, 1\}, \quad \forall (i, k, p). \tag{4}$$

The first represents a boundary on the amount of bandwidth that can be reserved on each link. The second and third represent the flow conservation constraint for each multicast flow i for the integer programming formulation of the problem: a connected tree from s_i to all the r_i receivers (identified by the ith column of Y) has to be set-up. This can be expressed as r_i unicast paths (3), each being a subset of the multicast tree (2) (for more details refer to Oliveira and Pardalos, 2005; Noronha and Tobagi, 1994).

The objective function is given by

$$\min \left[\alpha \sum_{i=1}^{L} \sum_{k=1}^{m} Y_{ki} b_i - \beta \min_{k=1,\ldots,m} \left(B_k - \sum_{i=1}^{L} Y_{ki} b_i \right) \right] \tag{5}$$

and thus the objective of the problem is twofold: (i) to minimise the amount of used network resources, i.e., allocated network bandwidth; (ii) to maximise the minimum amount of unallocated bandwidth within the backbone, so as to limit potential service blocking for future requests. The constant, positive values α and β represent the relative weights given to the two objectives.

Table 1
Multicast QoS routing problem: system parameters

Parameter	Description	
Inputs		
B	Vector (m)	Vector of size m, whose values are the maximum capacities associated with the edges
M	Matrix ($n \times m$)	The node–edge incidence matrix of size $n \times m$, equal to $M_{pk} = \begin{cases} +1 & \text{if node } p \text{ is the source of edge } k, \\ -1 & \text{if node } p \text{ is the sink of edge } k, \\ 0 & \text{otherwise.} \end{cases}$
L	Scalar	Number of multicast flows to support (indexed by i)
s_i	Scalar	Source node of the ith flow
b_i	Scalar	Amount of bandwidth required by the ith flow
r_i	Scalar	Number of receivers associated with the ith flow
n	Scalar	Number of nodes of the core network (indexed by p)
m	Scalar	Number of oriented edges of the graph (indexed by k)
A^i	L matrices ($n \times r_i$)	Matrix of size $n \times r_i$, whose columns represent the r_i source–destination couples of the ith multicast tree: $A^i_{pq} = \begin{cases} +1 & \text{if node } p \text{ is } s_i, \\ -1 & \text{if node } p \text{ is the } q\text{th receiver among the } r_i \text{ receivers,} \\ 0 & \text{otherwise.} \end{cases}$
Outputs		
Y	Matrix ($m \times L$)	Matrix of size $m \times L$, whose columns represent the multicast flows allocation on G, i.e., $Y_{ki} = 1$ if the edge k belongs to the ith multicast tree, $Y_{ki} = 0$ otherwise
X^i	L matrices ($m \times r_i$)	Matrix of size $m \times r_i$, whose columns represent a number of r_i unicast, directed paths from s_i to the r_i receivers. These unicast paths are subsets of the multicast path constituted by the ith column of the matrix Y, i.e., $X^i_{kp} = \begin{cases} 0 & \text{if } Y_{ki} = 0 \text{ OR edge } k \notin \text{unicast path connecting } s_i \text{ to the } p\text{th receiver of the } r_i \text{ receivers,} \\ 1 & \text{if } Y_{ki} = 1 \text{ AND edge } k \in \text{unicast path connecting } s_i \text{ to the } p\text{th receiver of the } r_i \text{ receivers.} \end{cases}$

According to the taxonomy of the multicast routing problems proposed in Wang and Hou (2000), our reference problem is clearly "link-constrained", as the constraints depend on the bandwidth availability at links (1).

It is worth noting that the QoS requirements, in terms of end-to-end delay, delay jitter, and losses, would imply, in general, additional constraints at *tree* level, and thus an additional complexity to the mathematical model. In this case, since we use a peak allocation strategy that can ensure network QoS guarantees (i.e., upper bounds on delay, jitter and losses), the complexity of the mathematical model is unchanged even with QoS support. For the sake of completeness, we have verified the achievement of the desired performance through simulations (see Section 4).

In addition, if $\beta = 0$, the objective function would classify the problem as a "tree optimisation" problem (i.e., Steiner tree problem); otherwise, if $\alpha = 0$, the objective function would classify the problem as a "link optimisation" problem (i.e., to locate an optimal multicast tree from a link-based objective function). Thus, if both α and β are positive, the considered problem is a "link-constrained link-and-tree optimisation problem".

Note that the constrained tree optimisation problems (commonly known also as constrained Steiner tree problems) have been proved to be NP-complete (Garey and Johnson, 1979).

It is also worth noting that the cost function is a nonlinear and non-convex function, and even professional/commercial tools cannot guarantee convergence to the global optimum.

In addition, due to the huge dimension of the admissible solutions space, it is unreasonable to search a solution by a brute-force approach. In fact, the cardinality of the variable t-uple set to be explored in order to find the solution, ranges within $[2^{(\mathrm{mL})(m \min\{r_i\}L)}, 2^{(\mathrm{mL})(m \max\{r_i\}L)}]$, where the first factor in the exponent (mL) represents the dimension of the Y solution space and the second factor in the exponent represents the boundaries for the dimension of the X solution space. For instance, if we consider the network scenario we have simulated and whose numerical results are presented in Section 4, we have $L = 36$ flows, and each flow has 8 destination hosts. Since the topology represents only the core network, the maximum number of receivers is 6 ($\max\{r_i\} = 6$), because there are 7 ingress–egress routers, one of which must be associated with the sender. The core network consists of 40 oriented edges ($m = 40$) and 10 nodes ($n = 10$). Thus, the cardinality of the variable t-uple set to be explored would range within $[2^{2073600}, 2^{12441600}]$.

In the following section, we present some alternative algorithms to find likely suboptimal solutions to our reference problem. Their performance, in terms of user-oriented and operator-oriented performance figures, will be deeply analysed in Section 4.

3.2. QoS-enabled multicast routing approaches

We consider three categories of approaches:

1. To find the solution of the original, complete problem by a commercial solver: the mathematical problem formulation (as presented in the previous Section 3.1) is encoded through the programming language of the solver, which should return a (sub)optimal solution.

2. To use algorithms based on the constrained Steiner tree strategy, able to solve the mathematical model with the objective function (5) with $\beta = 0$. We use such a strategy since it can solve a class of problems (constrained tree optimisation problems), which are similar to the proposed one.
3. To use algorithms based on the shortest-path tree strategy. They can solve tree constrained problems Wang and Hou (2000), which are a class of problems different from the proposed one. Nevertheless, we consider them since they are mostly used in practice by multicast routing protocols due to their ease of implementation and computation efficiency (Sahasrabuddhe and Mukherjee, 2000).

In more detail, the algorithmic approaches we consider are:

- use of the solver LINGO (LINGO tool) to find a solution of the original problem (category 1);
- synchronous optimisation (category 2);
- asynchronous optimisation (category 2);
- multicast near node first (category 2);
- Dijkstra (category 3);
- Dijkstra with TE support (category 3).

3.2.1. Original problem solution

This approach aims at using the commercial optimisation tool LINGO (LINGO tool) to solve the original problem defined in Section 3.1. Due to the nature of the problem, we expect that the solver could find a local optimum only. In the following, we will refer to this approach as LS.

3.2.2. Synchronous optimisation

The Synchronous Optimisation (Opt_S) approach aims at solving the problem defined in Section 3.1 with a simplified objective function (5). We set the value of α to 1 and the value of β to 0. In this way, the objective function becomes linear and a commercial tool such as LINGO is able to find the global optimum of the simplified problem. This approach consists in finding the Steiner tree for all service requests simultaneously. Hence, we name it "synchronous".

In order to take into account the second objective of the original problem (i.e., to maximise the minimum amount of unallocated bandwidth within the backbone), we may use a trick which allows maintaining the objective function linear: we subtract a bandwidth value equal to guard residual bandwidth (grB, whose value has to be lower than the bandwidth bottleneck of the network) from the speed of each link. Clearly, in this way we set a static, lower bound of the minimum residual bandwidth, thus forcing the algorithm to route streams away from links close to saturation. Thus, constraint (1) becomes $\sum_{i=1}^{L} Y_{ki} b_i \leqslant B_k - grB, \quad \forall k = 1, \ldots, m$, whereas all other constraints ((2)–(4)) remains unchanged.

3.2.3. Asynchronous optimisation

The Asynchronous Optimisation (Opt_A) approach is the asynchronous version of the approach illustrated in Section 3.2.2. Service requests are considered one by one consecutively, thus each multicast tree computation is influenced by the previously allocated multicast trees. This approach results in using LINGO to compute the Steiner tree for each request, thus the number of iterations is equal to L.

We stress that, at the tth iteration of the algorithm, constraints (1)–(4) of the original problem become

$$Y_{kt}b_t \leqslant B_k - grB - \sum_{i=1}^{t-1} Y_{ki}b_i, \quad \forall k = 1, \ldots, m, \tag{6}$$

$$X^t_{kp} \leqslant Y_{kt}, \quad \forall k = 1, \ldots, m, \quad \forall p = 1, \ldots, r_t, \tag{7}$$

$$MX^t = A^t, \tag{8}$$

$$X^t_{kp}, Y_{kt} \in \{0, 1\}, \quad \forall (k, p). \tag{9}$$

The objective function becomes

$$\min \left[\sum_{k=1}^{m} Y_{kt}b_t \right]. \tag{10}$$

3.2.4. *Multicast near node first (MNF) algorithm*

Multicast near node first (MNF) (Kodialam et al., 2003) is a directed, Steiner tree computation algorithm, which works on the subgraph that includes only those links that have residual capacity greater than or equal to b_i (requested flow rate). This is done by means of heuristics based on Dijkstra's shortest-path algorithm. The algorithm was developed for online routing of bandwidth-guaranteed multicast requests. The term "online" means that the multicast routing requests are handled asynchronously one at a time, without any awareness of future requests.

This approach results in using the MNF routine to compute the Steiner tree for each request, and thus the number of iterations is equal to L.

In order to compute the Steiner tree, the MNF approach needs to associate a weight (i.e., a cost), w_k, with each link k, which is different from the flow bandwidth b_i. The link weights are computed on the basis of the concept of criticality. Links are defined "critical" when loading these links causes a reduction in the multicast flow between an ingress and certain subsets of egresses. This causes "interference" to the capacity available for routing future demands. MNF tries to "minimally interfere" with paths needed for future demands and this is carried out by deferring loading of critical links as far as possible. The goal is to set-up P2MP LSPs with bandwidth guarantees and to exploit the knowledge of the ingress–egress LSRs in order to minimise the number of rejected receivers (those which cannot be added to a multicast tree due to link capacity limitations).

For further details, the interested reader should refer to Kodialam et al. (2003) and references therein.

At the tth iteration of the algorithm, the constraints (1)–(4) of the original problem become

$$Y_{kt}b_t \leqslant B_k - \sum_{i=1}^{t-1} Y_{ki}b_i, \quad \forall k = 1, \ldots, m, \tag{11}$$

$$X_{kp}^t \leqslant Y_{kt}, \quad \forall k = 1, \ldots, m, \quad \forall p = 1, \ldots, r_t, \tag{12}$$

$$MX^t = A^t, \tag{13}$$

$$X_{kp}^t, Y_{kt} \in \{0, 1\}, \quad \forall (k, p). \tag{14}$$

The objective function becomes

$$\min \left[\sum_{k=1}^{m} Y_{kt} w_k \right]. \tag{15}$$

We stress that such an objective function can address implicitly both the targets of our reference model: (i) to minimise the usage of network resources; (ii) to maximise the minimum amount of unallocated bandwidth within the backbone.

3.2.5. The Dijkstra algorithm

The basic Dijkstra algorithm is a well-known shortest-path algorithm, which minimises the sum of the weights on the links along each unicast path from a source to any receiver Cormen et al. (2001). It is possible to build a shortest-path multicast tree by running the Dijkstra algorithm for each source–receiver pair of the multicast group. We stress that there is no control on bandwidth availability by the routing algorithm. Clearly, this approach results in using the Dijkstra routine to compute the shortest-path tree for all receivers for each multicast stream, thus the number of iterations is equal to L.

Once the Dijkstra algorithm has computed the multicast trees, we have considered three different ways of operation to set-up the P2MP LSPs in the MPLS backbone:

- Basic Dijkstra (D_noRes), where the P2MP LSPs are set-up by RSVP-TE without any bandwidth reservation.
- Dijkstra with bandwidth reservation and Integrity bit set (D_Res_I), where the P2MP LSPs are set-up by RSVP-TE with bandwidth reservation and, in the case of missing resources for a part of a multicast tree, the set-up of the overall P2MP LSP associated with that tree fails (Aggarwal et al., 2007).
- Dijkstra with bandwidth reservation and Integrity bit unset (D_Res), where the P2MP LSPs are set-up by RSVP-TE with bandwidth reservation and, in the case of missing resources for a part of a multicast tree, only the set-up of that part of the tree fails.

3.2.6. The Dijkstra TE algorithm

The Dijkstra TE (D_TE) algorithm is a shortest-path algorithm, which adds a preliminary control on the available bandwidth of the links: all links whose available bandwidth is less than the flow rate, b_i, are removed from the shortest-path tree computation. This information can be made available by OSPF-TE signalling messages Katz et al. (2003). This translates into the additional constraint (6) with respect to the classic shortest-path problem formulation for the tth iteration of the algorithm. Since the Dijkstra TE algorithm is asynchronous with respect to the different requests, each multicast tree computation is influenced by the previously allocated multicast trees.

4. Performance evaluation

In this section, we first give a sketch of the software platform used to carry out the numerical analysis. Then we describe the network scenario of the simulation, and finally we illustrate an extensive comparative analysis of the multicast QoS algorithms described in the previous section, accounting for a number of performance figures.

4.1. Performance evaluation tools

The simulation platform used to perform the numerical analysis is INET. The INET Framework is built upon OMNeT++, an object-oriented, modular, discrete event network simulator (INET Framework for OMNeT++/OMNEST). We began with the basic simulator core and included all modules able to manage bandwidth-guaranteed P2MP LSPs with the support of RSVP-TE. The configuration files used by the enhanced INET (E_INET) module to set the optimal multicast trees may be created by starting from the output files of the optimisation tool. Then, the output files of E_INET will be elaborated using Matlab and/or MS Excel to get performance curves.

The overall simulation environment is depicted in Fig. 7.

We used different optimisation tools to solve the problem, depending on the solution strategy adopted. In particular, LS, the Synchronous and Asynchronous Optimisation algorithms were implemented using LINGO 8.0 (LINGO tool), whereas the MNF algorithm was developed in C++. LINGO is a tool for formulating and solving large linear and nonlinear optimisation problems. As for the Dijkstra-based algorithms, these are implemented by an INET module.

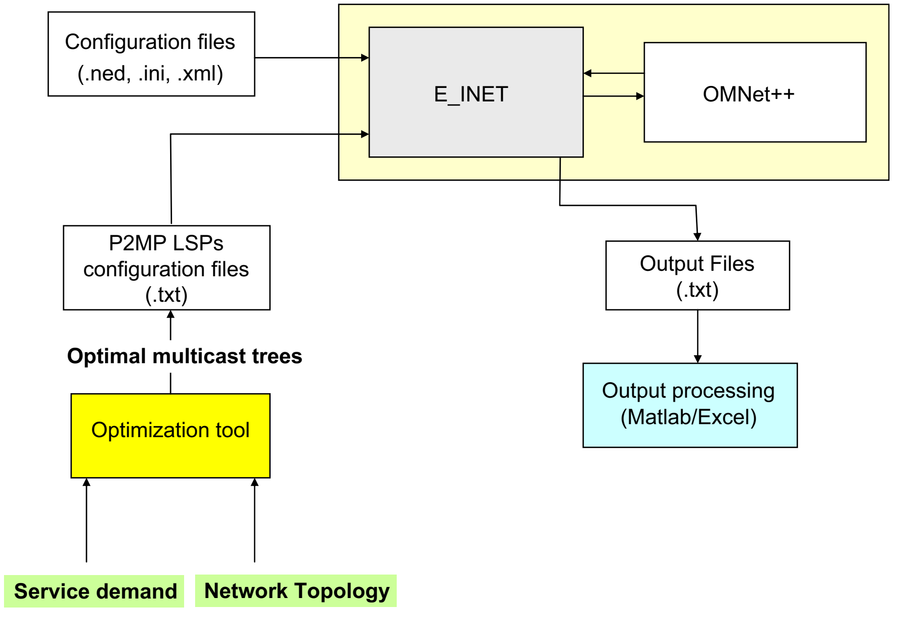

Fig. 7. Performance evaluation system: an overall picture.

4.2. Simulated network scenario

The simulated network is represented in Fig. 8; it consists of 28 hosts, 7 access routers (LSR1, LSR2, LSR3, LSR5, LSR6, LSR7, LSR10), and 3 core routers (LSR4, LSR8, LSR9). The speed of access links are 2488.32 Mb/s (OC-48 with 2405.376 Mb/s of payload bandwidth). This choice was dictated by the need to avoid bottlenecks in the access network. As for the backbone, all links have a propagation delay equal to 1 ms and a capacity equal to:

- 2488.32 Mb/s (OC-48 with 2405.376 Mb/s of payload bandwidth) for links connecting core routers (e.g., links among LSR4, LSR8, and LSR9).
- 155.52 Mb/s (OC-3, 148.608 Mb/s of payload bandwidth) for the link between LSR4 and LSR10.
- 622.08 Mb/s (OC-12 with 601.344 Mb/s of payload bandwidth) for other links.

There are 36 multicast data flows, each one associated with a source (server application), a multicast group (8 client applications), a multicast address and a bit rate. As regards the bit rate, there are two different values at application layer (20 and 50 Mb/s, as discussed in Section 2.3), depending on the coding scheme chosen by the Exhibitor, i.e., according to the capacity of the access network and/or the projection equipment. We assumed that flows are constant bit rate (CBR). Table 2 reports the different multicast streams with source and destination hosts. The flows with an odd ID have a rate equal to 20 Mb/s, whereas the others are equal to 50 Mb/s. Each live event is modelled with two different

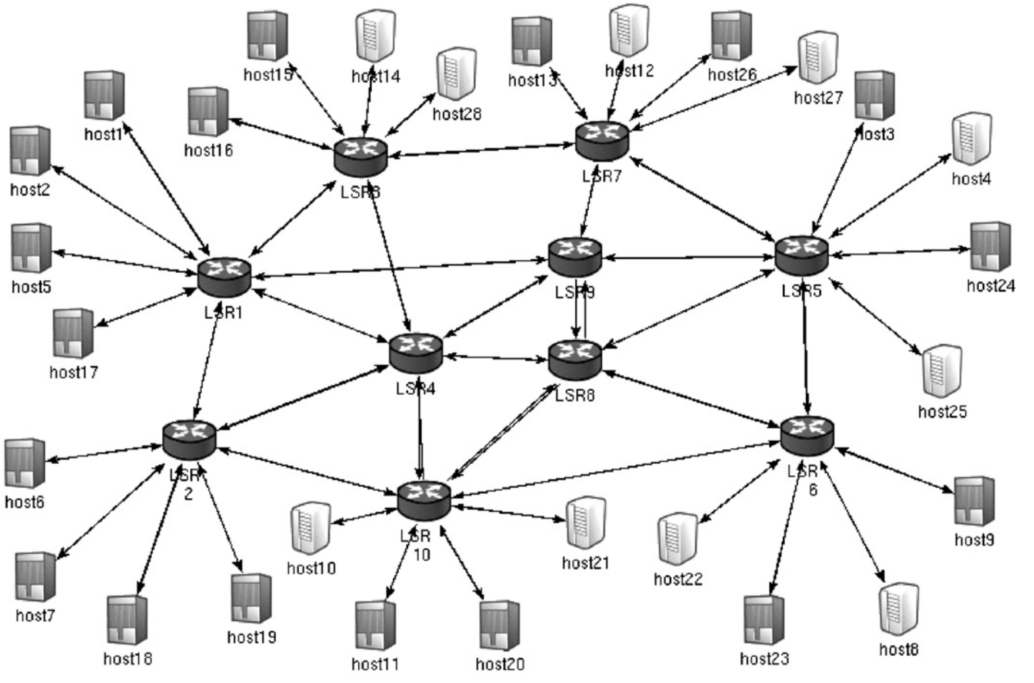

Fig. 8. Simulated network.

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

Table 2
Multicast streams

ID	Source	Destinations
1	host1	host2-host5-host7-host8-host11-host15-host22-host25
2	host1	host3-host4-host6-host9-host10-host13-host16-host24
3	host2	host1-host5-host7-host10-host13-host15-host20-host24
4	host2	host6-host8-host9-host12-host14-host17-host22-host27
5	host6	host1-host5-host9-host12-host16-host17-host21-host28
6	host6	host3-host4-host8-host10-host14-host19-host24-host27
7	host7	host1-host6-host10-host17-host19-host23-host25-host27
8	host7	host5-host8-host14-host18-host21-host24-host26-host28
9	host11	host1-host3-host6-host9-host13-host18-host20-host26
10	host11	host2-host4-host7-host12-host16-host19-host24-host28
11	host9	host2-host3-host5-host7-host14-host22-host24-host26
12	host9	host1-host4-host6-host10-host12-host20-host23-host25
13	host3	host1-host5-host6-host8-host10-host14-host19-host21
14	host3	host2-host4-host7-host9-host11-host13-host16-host20
15	host13	host2-host3-host7-host12-host16-host17-host22-host26
16	host13	host1-host5-host9-host11-host14-host18-host21-host23
17	host15	host1-host2-host6-host10-host11-host13-host23-host27
18	host15	host3-host4-host9-host12-host16-host17-host24-host28
19	host16	host1-host5-host7-host10-host12-host17-host21-host26
20	host16	host2-host4-host8-host11-host13-host15-host19-host25
21	host20	host1-host2-host4-host7-host16-host17-host21-host26
22	host20	host3-host6-host9-host11-host18-host19-host24-host27
23	host23	host1-host3-host5-host11-host14-host16-host19-host25
24	host23	host4-host8-host9-host13-host15-host18-host22-host27
25	host24	host1-host4-host6-host13-host16-host18-host20-host27
26	host24	host2-host3-host9-host10-host14-host17-host19-host23
27	host26	host3-host4-host11-host13-host18-host21-host23-host28
28	host26	host1-host5-host10-host12-host16-host19-host22-host25
29	host5	host2-host4-host7-host11-host13-host17-host23-host27
30	host5	host3-host6-host8-host10-host12-host16-host19-host24
31	host17	host1-host3-host6-host9-host11-host16-host18-host20
32	host17	host2-host4-host5-host8-host13-host14-host19-host21
33	host18	host1-host3-host4-host7-host10-host13-host16-host25
34	host18	host2-host5-host8-host9-host11-host14-host17-host28
35	host19	host1-host4-host6-host9-host12-host17-host22-host24
36	host19	host3-host5-host7-host11-host14-host21-host23-host27

flows, one at 20 Mb/s and the other at 50 Mb/s. Their multicast groups are disjointed. The size of the packet payload is equal to 1400 B, slightly below the Ethernet maximum packet length (1500 B), and the size of the headers is equal to 39 B (UDP, IP, MPLS and PPP headers).

Each simulation lasts 350 s, and each flow starts at time 5 s and stops at 300 s. The P2MP LSPs set-up starts at time 1 s and last a few milliseconds, so that at time 5 s all P2MP LSPs are up. It is worth noting that in order to extract significant statistics, simulation of the whole duration of the live event (e.g., 2 h for a sports event) is unnecessary and a few minutes will suffice, as the flows are CBR.

In the physical output interfaces of LSRs, each P2MP LSP is associated with a bandwidth-guaranteed, logical queue with a buffer size equal to 20 packets. The scheduling discipline adopted to manage multiple logical queues is weighted fair queuing (WFQ) (Demers et al., 1989).

In order to satisfactorily support the live event delivery service, we expect values of end-to-end delay below 1 s, negligible (next to zero) packet loss rates and values of delay jitter in the order of tens of milliseconds.

4.3. Numerical results

The performance figures we have evaluated for all the approaches solving the multicast QoS routing problem are:

- User-oriented QoS parameters: end to end delay, delay jitter, packet losses and number of unreachable receivers.
- Operator-oriented network management parameters: total traffic handled by the core network, minimal residual bandwidth, operational cost in terms of network resources, computation time and RSVP-TE signalling traffic overhead.

With reference to the cost function (5) of the original problem described in Section 3.1, we consider three different, meaningful couples of values for α and β (selecting $\alpha = 1/\beta$):

- $\alpha = 1 \textit{and}\ \beta = 1$: This means that the first component of (5), $\textit{Traffic} = \sum_{i=1}^{L}\sum_{k=1}^{m} Y_{ki} b_i$ (the total traffic handled by the core network), is dominant with respect to the second component, i.e., the minimum residual bandwidth, $MRB = \min\limits_{k=1,\ldots,m}\left(B_k - \sum_{i=1}^{L} Y_{ki} b_i\right)$. In fact, we expect the first component to be several two orders of magnitude higher than the second. This implies that the target of the original optimisation problem is mainly to minimise *Traffic*.
- $\alpha = 0.1\ \textit{and}\ \beta = 10$: This configuration should balance the two contributions, i.e., the original optimisation problem should both minimise *Traffic* and maximise *MRB*.
- $\alpha = 0.01\ \textit{and}\ \beta = 100$: With this configuration, the second component should be dominant, and thus the target of the original optimisation problem is mainly to maximise *MRB*.

4.3.1. Numerical results: user-oriented QoS performance figures

As regards packet losses, as expected, only the D_noRes approach (the only one without resource reservation) experiences packet losses; the average packet loss is approximately 5% and the maximum packet loss experienced by a multicast stream is equal to 79%. Regarding the number of blocked receivers, only D_Res_I and D_Res have unreachable receivers. This is due to the fact that these two approaches exploit the Dijkstra route computation without performing a previous check on available resources. Clearly, since the D_Res_I does not allocate the whole P2MP LSP, if at least one of the receivers cannot be reached by the multicast stream, it has a higher value of unreachable receivers (24.65%) than D_Res (5.9%). In the case of D_Res_I, for a multicast stream with a failed P2MP LSP set-up procedure, the only reached hosts are those having the same access router as the relevant server host. This occurs because in this case packets are forwarded at IP level (the P2MP LSP is not set-up).

The straightforward comment is that the Dijkstra route computation without accounting for the amount of available resources can be used only if the backbone is strongly over-provisioned. Thus, D_noRes, D_Res_I and D_Res are not good candidates for solving the multicast QoS routing problem.

The (maximum and average) edge-to-edge (e2e) delay values are shown in Fig. 9. This parameter is the ingress–egress backbone delay.

As expected, the worst performance is achieved by the D_noRes approach, since some links prove to be overloaded. The D_Res_I and D_Res approaches work very well, but we need to bear in mind that some clients are not served (see the previous comments on blocked receivers). The D_TE performance is slightly better than Opt_S, Opt_A and LS since the latter approaches mainly select the outer ring links (those between the egress nodes), in order to minimise the overall hops of the P2MP LSPs. In these links, packets experience a higher queuing delay because the amount of traffic is larger. Instead, the D_TE approach minimises the distance between the source and each destination, thus experiencing a lower delay. For the D_TE, Opt_S and Opt_A approaches, the delay proves to be independent of the value of *grB*. Finally, the delay value associated with MNF is greater than that of the D_TE due to the fact that MNF does not try to minimise the delay or to balance the load over the links; in fact, its main goal is to avoid using the critical links that may interfere with future demands.

As a general comment, we can conclude that all the approaches provide e2e delay values, which are compliant with the live event delivery service.

A parameter linked to the e2e delay is the diameter of a multicast tree, defined as the maximum number of links between the ingress–egress couple. The values of mean and maximum diameter for all the multicast trees of all approaches are shown in Fig. 10. As expected, the approaches based on Dijkstra give similar results, which are the best due to the nature of the Dijkstra algorithm. Clearly, D_Res presents a lower mean value due to the pruning mechanism. The other approaches have a similar performance.

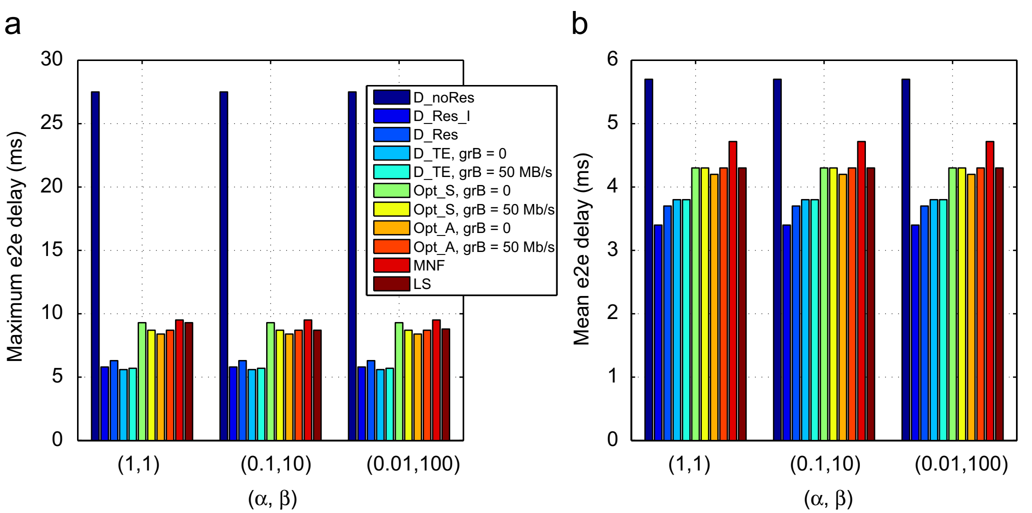

Fig. 9. Edge-to-edge delay: (a) max value and (b) mean value.

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

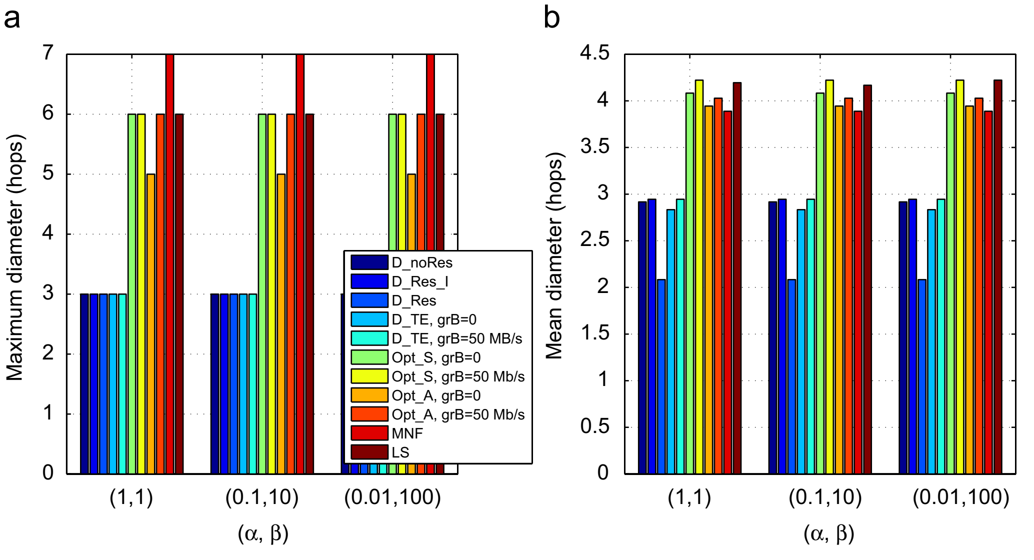

Fig. 10. Diameter: (a) max value and (b) mean value.

The (maximum and average) delay jitter values are presented in Fig. 11. Here, we define the delay jitter as the standard deviation of the e2e delay. The comments relevant to the delay jitter values are similar to those of the e2e delay analysis. As a general comment, we can conclude that all the approaches provide a delay jitter, which is fully compliant with the live event delivery service (tens of ms). As for Dijkstra without resource reservation, note that this compliance is mainly due to the fact that each P2MP LSP is associated with a logical queue with a buffer size equal to 20 packets only at each router; in fact packet losses are quite high. For all the other approaches, low values of delay jitter are due to peak rate allocation within logical queues.

4.3.2. *Numerical results: operator-oriented performance figures*

As for the total traffic handled by the core network, the values of the overall gross rate (*Traffic*) associated with the various approaches are reported in Fig. 12a. Let us remember that this parameter is related to all the links forming the P2MP LSPs, which support the delivery of multicast streams. For the first three approaches (D_noRes, D_Res_I and D_Res), the P2MP LSPs are identical, since they are computed by means of the Dijkstra algorithm without taking into account available bandwidth information. What leads to different traffic values is the way in which the resource allocation is managed, as discussed in detail above. The result is that:

- D_noRes has a traffic value close to 1.13 GB/s, with packet losses due to queue overflow.
- D_Res has a traffic value close to 1.10 GB/s, and some packets are not forwarded since some branches of P2MP LSPs are pruned.
- D_Res_I has a slightly lower amount of traffic than previous cases (about 0.80 GB/s), since some P2MP LSPs are not completely set-up.

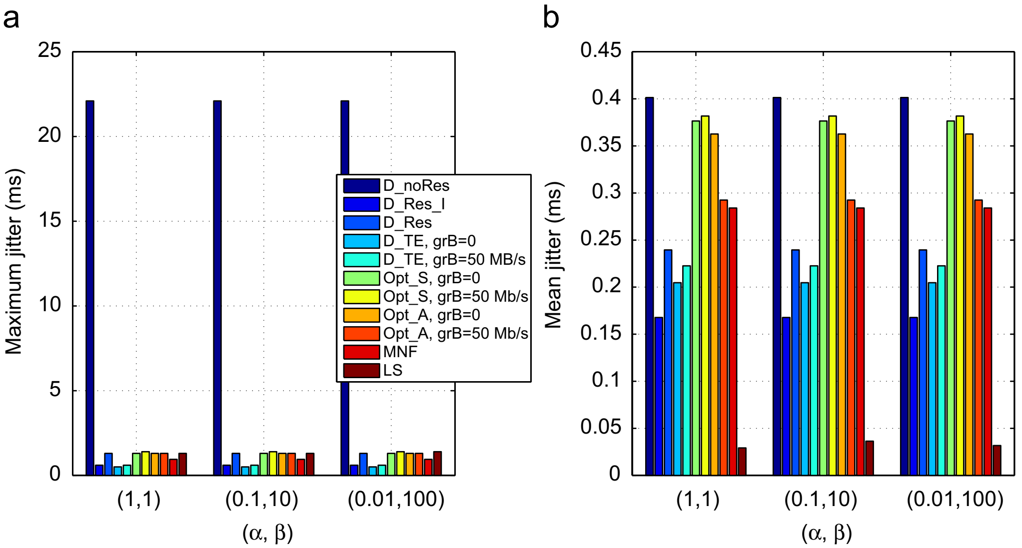

Fig. 11. Delay jitter: (a) max value and (b) mean value.

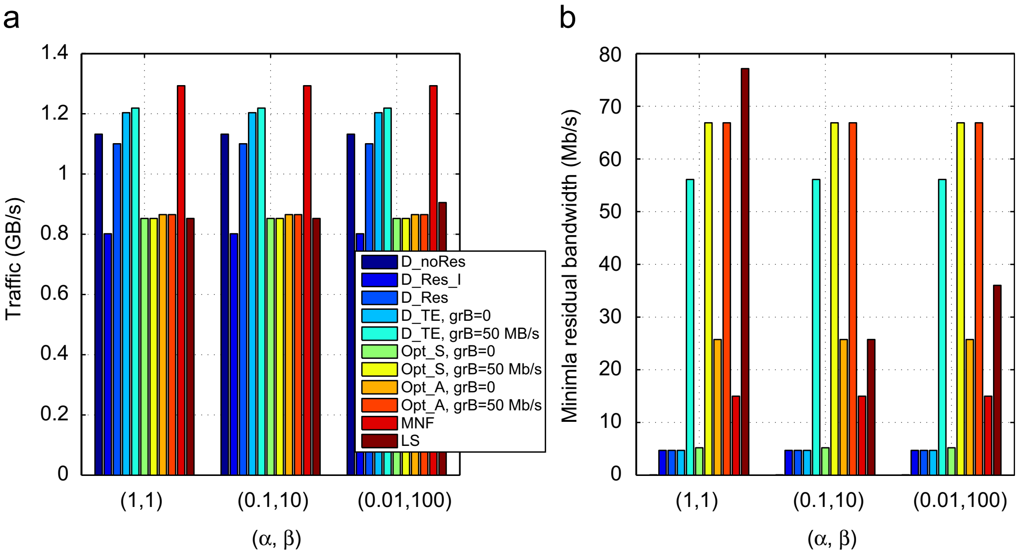

Fig. 12. (a) Total traffic handled by the core network and (b) minimal residual bandwidth.

The D_TE approach has the same basic multicast tree computation approach (Dijkstra). However, the initial bandwidth availability check enables all receivers to be reached by the flow. This means that D_TE has a greater value of total, handled traffic (approximately 1.2 GB/s) than the previous approaches. As for the MNF algorithm, it performs slightly worse than D_TE, since it computes the multicast tree as a Steiner tree, trying to avoid potentially critical links. This translates in a choice of P2MP LSPs, which may span further over the network, thus involving more links. As regards Opt_S and Opt_A, they focus on

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

total bandwidth minimisation, and thus provide the best performance. Their gain with respect to D_TE and MNF is approximately 30%. Due to its synchronous nature, Opt_S slightly outperforms Opt_A. Again, the value of *grB* does not affect the performance of D_TE, Opt_S and Opt_A.

Finally, LS is characterised by the same performance of Opt_S for the first two (α,β) configurations ((1,1) and (0.1,10)). For this reason, LS also tries to minimise the *Traffic* component for these configurations. For the other configuration ($(\alpha,\beta) = (0.01,100)$), since the target of LS is mainly to maximise *MRB*, the amount of *Traffic* is higher than in the previous two cases.

In Fig. 12b, *MRB* is reported. *MRB* is an indicator of the capability of the network to route new QoS-guaranteed LSPs, and thus the higher MRB is, the more the probability of service blocking is reduced. Simulations show that D_noRes experiences a value close to zero, since neither a bandwidth check nor a reservation are executed, and thus some interfaces are overloaded. For D_Res_I and D_Res the value is low and close to 5 Mb/s (lower than used flow data rates). Note that these approaches also present a number of unreachable receivers (and thus lower values of handled traffic). D_TE and Opt_S have similar values with respect to the previous two, but are able to deliver a higher amount of traffic (no receivers are blocked). This behaviour is due to the capability of setting up P2MP LSPs according to the amount of available resources within core links. The low value of *MRB* is justified by objective functions, which do not account for residual bandwidth. On the contrary, the MNF algorithm, as expected, presents a better performance (*MRB* around 14 Mb/s). It is surprising that Opt_A shows further improvement with approximately 25 Mb/s remaining, even though such an approach does not account for the residual bandwidth in its objective function. The result is probably due to the fact that Opt_A sets up P2MP LSPs, which are different with respect to Opt_S, and this choice leads to larger values of *Traffic* and *MRB*.

The simulations for D_TE, Opt_S and Opt_A are repeated with $grB = 50$ Mb/s for each link. The motivation of this choice is that, in this way, we can thus ensure enough capacity in core routers to accept at least an additional multicast flow at the highest rate. In this case, the three approaches are able to find an admissible solution, and thus the *MRB* is higher than 50 Mb/s. Note that Opt_S and Opt_A have very similar values (about 67 Mb/s), even though the P2MP LSPs are not the same (due to different *Traffic* values, see Fig. 12a). The value associated with the D_TE approach is lower, and about 56 Mb/s.

A final comment concerns the LS approach. For the configuration $(\alpha,\beta) = (1,1)$, it not only minimises *Traffic*, but also provides the best value for *MRB*, equal to 77 Mb/s. This is an expected result, since this configuration, which aims mainly to minimise the handled traffic, should also minimise (with lower weight) the *MRB*. However, since the other two configurations give a higher importance to the *MRB* contribution, we expected at least a similar behaviour. On the contrary, in these two cases the LINGO solver is unable to move away from a local optimum. Anyway, the performance of the third configuration (36 Mb/s for $(\alpha,\beta) = (0.01,100)$) is better than the second (25.7 Mb/s for $(\alpha,\beta) = (0.1,10)$), as expected.

Now, let us consider the total operational cost (in terms of network resources) for each approach. Such a cost is defined in accordance with the objective function of the optimisation problem defined in (5), i.e., the weighted sum of the amount of used network resources, *Traffic* (reported in Fig. 12a), and of *MRB* (reported in Fig. 12b):

$$Cost = \alpha \times Traffic - \beta \times MRB. \tag{16}$$

Here, we consider as candidate approaches those which are able to satisfy all the QoS requirements, i.e., Dijkstra TE, Opt_S, Opt_A, MNF and LS.

For the three configurations of α and β, we expect definitely positive cost values for (1,1), cost values close to zero for (0.1,10), and definitely negative cost values for (0.01,100). Table 3 reports the cost values for all the candidate approaches, normalised by 1 Mb/s. For each (α,β) configuration, the minimum value is highlighted in bold type.

The main comment is that the best performing approach is the Opt_S with $grB = 50$ Mb/s. In fact, it outperforms all the others for both $(\alpha,\beta) = (0.1,10)$ and $(\alpha,\beta) = (0.01,100)$. In addition, for $(\alpha,\beta) = (1,1)$, performance is really very close to that of LS.

The values of computation time for the considered simulation scenario and for all approaches are summarised in Table 4. All approaches have a computation time compliant with the request-routing procedure. As for the Dijkstra and Dijkstra_TE algorithms, they are embedded in the E_INET simulator, and if we observe the simulation logs, their execution times prove negligible. The highest computation time is that of LS and is equal to approximately 1 h. Thus, if LINGO is adopted to solve the original problem, Exhibitors will have to book live events several hours in advance of the event start. This is a reasonable constraint, because the projection of events has to be advertised in advance. However, in this case also, we are confident that the computation time can be reduced by using a powerful workstation.

Table 3
Operational cost: comparison among the candidate approaches

Approach		Operational cost		
		$\alpha = 1$, $\beta = 1$	$\alpha = 0.1$, $\beta = 10$	$\alpha = 0.01$, $\beta = 100$
Dijkstra TE	$grB = 0$	9623.294	915.740	−374.320
	$grB = 50$ Mb/s	9695.100	414.120	−5512.488
Synchronous optimisation	$grB = 0$	6814.013	630.050	−450.508
	$grB = 50$ Mb/s	6752.344	**13.360**	**−6617.408**
Asynchronous optimisation	$grB = 0$	6895.855	434.710	−2505.284
	$grB = 50$ Mb/s	6854.743	23.590	−6616.484
MNF		10,329.814	884.620	−1395.152
LS		**6742.000**	424.7	−3511.46

Table 4
Computation time (with a standard PC)

Approach	Solver	Computation time
Dijkstra/Dijkstra TE	E_INET	<1 s
Synchronous optimisation	LINGO	~10 s
Asynchronous optimisation	LINGO	~10 s
MNF	Implementation in C++	~90 s
LS	LINGO	~1 h

Finally, Table 5 illustrates the RSVP-TE signalling overheads associated with the different approaches. With respect to the data flows whose rate is in the order of tens of Mb/s, these values are definitely negligible in all cases; in fact, they are lower than 31.63 KB/s, which is the value experienced by D_TE due to the greater number of used links.

4.3.3. Discussion

To sum up, the Opt_S approach, which solves a simplified version of the original problem, presents the best performance. In fact, in the network scenario analysed, it guarantees the QoS level will support the live event delivery service, it minimises the overall traffic in the core network, it is able to control the amount of *MRB* with the use of the *grB* parameter, and it provides the best performance in terms of operational cost with a very low computation time.

Since the computation time of Opt_S is really low, it is possible to refine the approach by running the algorithm more times, using values of *grB* spanning from zero to the minimum value of link capacity. Then, the final choice of *grB* is the one which minimises the operational cost, and the set of P2MP LSPs to deploy is the one associated with such a value.

In this regard, Fig. 13 reports the values of the Opt_S cost (16) versus *grB* ranging from 0 to 140 Mb/s for different values of (α,β). We highlight that these values have to be defined by the NSP according to proprietary policy; it may move the optimisation problem towards either the minimisation of the overall traffic (high values of α) or the maximisation of the minimal residual bandwidth (high values of β). We also report the values of *Traffic* and *MRB* versus *grB* in Fig. 14.

For the cases with $(\alpha,\beta) = (2,0.5)$ and $(\alpha,\beta) = (1,1)$, which give more importance to the minimisation of *Traffic*, the best value of *grB* is equal to 80 Mb/s. In fact, *Traffic* does not increase up to that value, whereas *MRB* does.

Table 5
RSVP-TE traffic

Multicast routing approach		RSVP-TE traffic (KB/s)
D_noRes		30.01
D_Res_I		29.93
D_Res		29.99
D_TE	$grB = 0$	31.63
	$grB = 50$ Mb/s	31.63
Opt_S	$grB = 0$	29.93
	$grB = 50$ Mb/s	29.93
Opt_A	$grB = 0$	29.93
	$grB = 50$ Mb/s	29.93
MNF		30.05
LS	$(\alpha,\beta) = (1,1)$	29.93
	$(\alpha,\beta) = (0.1,10)$	29.93
	$(\alpha,\beta) = (0.01,100)$	29.94

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

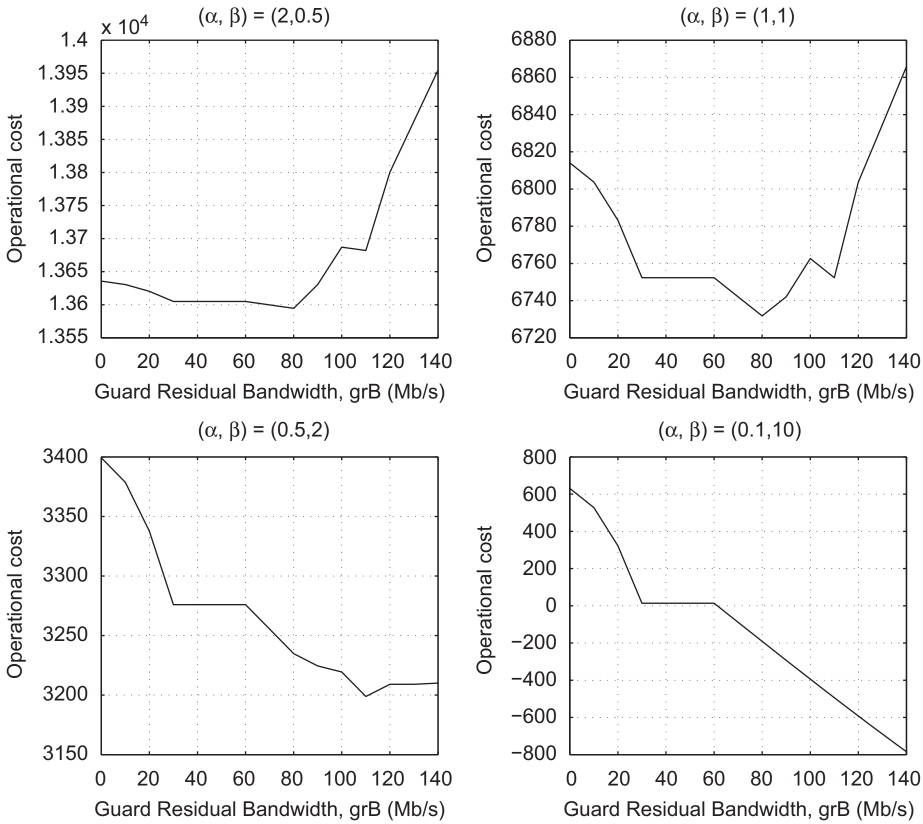

Fig. 13. Opt_S cost versus *grB* for different values of (α,β).

For the case with $(\alpha,\beta) = (0.5,2)$, where the maximisation of *MRB* is stressed, the best choice for *grB* is equal to 110 Mb/s. In fact, for values of *grB* higher than 110 Mb/s the slope of the *Traffic* curve increases sharply, whereas the *MRB* slope is nearly constant.

Finally, for the case with $(\alpha,\beta) = (0.1,10)$, the maximisation of *MRB* is definitely foregoing and thus the minimum value is obtained for *grB* equal to the extreme value (140 Mb/s).

5. Conclusion

In this paper, we have presented some key issues related to the distribution of DC contents via a Content Delivery Network.

We have especially focussed on the live event delivery within an MPLS backbone. We have analysed in detail the multicast QoS routing problem associated with the transport of large streams towards a set of theatres. We have first presented the problem from the architectural and procedural viewpoints, and then we have moved towards the algorithmic aspects. We have presented the mathematical model, and then a number of different approaches to solve it, and compared them by means of an extensive simulation campaign performed with an extended version of the OMNeT++ simulation platform. We have

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004

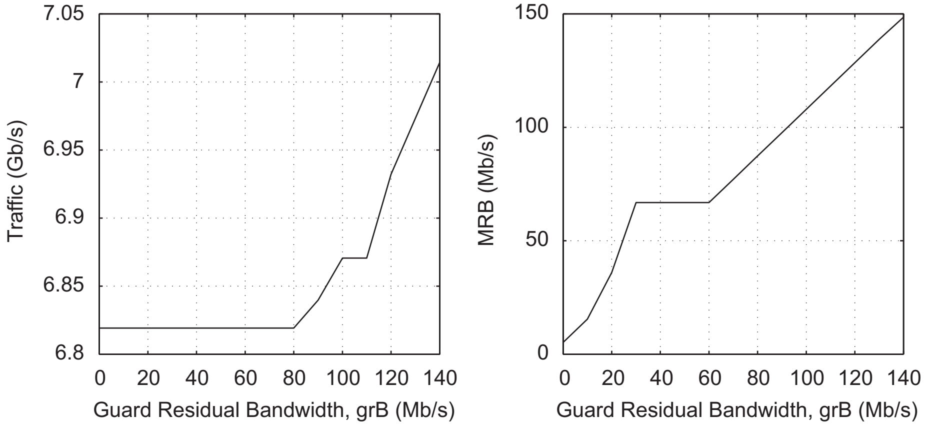

Fig. 14. Opt_S *Traffic* and *MRB* versus *grB*.

evaluated not only the operational cost (in terms of network resources), but also user-side QoS parameters (packet losses, end-to-end delay, delay jitter, and service blocking) and operator-side network management parameters (total traffic handled by the core network, minimal residual bandwidth, and RSPV-TE traffic overhead).

We can conclude that, even though the nature of the objective function of the multicast QoS routing problem does not guarantee the convergence to a global optimum, it is possible to achieve a good solution by using the Opt_S approach, which solves a reduced version of the original problem. In fact, in the network scenario analysed, Opt_S is able to

- guarantee the QoS level to support the live event delivery service;
- minimise the overall traffic in the core network;
- control the amount of *MRB* by varying the *grB* parameter;
- provide the best performance in terms of operational cost;
- provide a solution with a very low computation time.

Acknowledgements

This work was performed under co-financing by the European Union within the Integrated Project IST-2005-038454 EDCINE (Enhanced Digital CINEma).

The effort of the other project partners is gratefully acknowledged. This paper, however, does not implicitly and necessarily represent their opinion.

The authors would like to thank Mrs. Agnese Grasselli and Mr. Alessandro Parisi for their valuable support in the simulation activity.

References

Aggarwal R, Papadimitriou D, Yasukawa S. Extensions to RSVP-TE for point-to-multipoint TE LSPs. IETF RFC 4875, May 2007.

Awduche D, et al. RSVP-TE: extensions to RSVP for LSP tunnels. IETF RFC 3209, December 2001.

Awduche D, Malcolm J, Agogbua J, O'Dell M, McManus J. Requirements for traffic engineering over MPLS. IETF RFC 2702, September 1999.

Bertella A, Mignone V, Sacco B, Tabone M. Laboratory evaluations of DVB-S2 state-of-the-art equipment. EBU Techn Rev 2007.

Bilgin A, Marcellin MW. JPEG2000 for Digital Cinema. In: ISCAS06, Kos Island, Greece, May 2006.

Cormen TH, Leiserson CE, Rivest RL, Stein C. Introduction to algorithms. Cambridge, MA: MIT Press; 2001.

Daniel A. IP virtual private networks—a service provider perspective. IEE Proc Commun 2004;151(1).

Demers A, Keshav S, Shenker S. Analysis and simulation of a Fair Queueing Algorithm. In: SIGCOMM 89, Austin, USA, September 1989.

Digital Cinema System Specification v1.1, May 2007, 〈http://www.dcimovies.com/press/DCI_Press_Release_3-May2007.pdf〉.

Duffield NG, et al. A flexible model for resource management in virtual private networks. In: SIGCOMM'99, August–September 1999, Cambridge, USA.

EDCF, 2003, Technical Mudule Report, 〈http://www.edcf.net/technical_docs/2003%20Report%20170204.pdf〉.

EDCINE Project Deliverable D1.1. Requirements and general conditions. Available at 〈http://www.edcine.org/documents/public/edcine_ucl_del_d11_d6_7_final_v4.pdf/view〉.

Garey MR, Johnson DS. Computer and intractability. New York: W.H. Freeman and Company; 1979.

INET Framework for OMNeT++/OMNEST, web page: 〈http://www.omnetpp.org/doc/INET/neddoc/index.html〉.

Information technology-JPEG 2000 image coding system—Part 1: Core coding system. Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 15 444-1:2000 and ITU-TT.800.

IST EDCINE Project Web Site, 〈http://www.edcine.org/〉.

ITU-T Recommendation H.264. Advanced video coding for generic audiovisual services, March 2005.

Katz D, Kompella K, Yeung D. Traffic engineering (TE) extensions to OSPF version 2. IETF RFC 3630, September 2003.

Kodialam M, Lakshman TV, Sengupta S. Online multicast routing with bandwidth guarantees: a new approach using multicast network flow. IEEE/ACM Trans Networking 2003;11(4):676–86.

LINGO tool, 〈http://www.lindo.com/products/lingo/lingom.html〉.

Morin T. Requirements for multicast in layer 3 Provider-Provisioned Virtual Private Networks (PPVPNs). IETF RFC 4834, April 2007.

Noronha CA, Tobagi FA. Optimum routing of multicast streams. In: IEEE INFOCOM'94, Toronto, Canada, June 1994.

Oliveira CAS, Pardalos PM. A survey of combinatorial optimization problems in multicast routing. Comput Oper Res 2005;32(8):1953–81.

Rosen E, Rekhter Y. "BGP/MPLS VPNs," IETF RFC 2547, March 1999.

Rosen E, Viswanathan A, Callon R. Multiprotocol Label Switching Architecture. IETF RFC 3031, January 2001.

Sahasrabuddhe L, Mukherjee B. Multicast routing algorithms and protocols: a tutorial. IEEE Network 2000 January/February 2000.

SMPTE Web Site, 〈http://www.smpte.org/〉.

The Virtual Private Network Consortium, 〈http://www.vpnc.org/〉.

Verma DC. Content distribution networks: An engineering approach. New York: Wiley; 2002.

Wang B, Hou JC. Multicast routing and its QoS extension: Problems, algorithms, and protocols. IEEE Network 2000 January/February 2000.

Yasukawa S. Signaling requirements for point-to-multipoint traffic-engineered MPLS label switched paths (LSPs). IETF RFC 4461, April 2006.

Please cite this article as: Di Sorte D, et al. QoS-enabled multicast for delivering live events in a Digital Cinema scenario. J Network Comput Appl (2008), doi:10.1016/j.jnca.2008.02.004