Network Delivery of Live Events in a Digital Cinema Scenario

Dario Di Sorte, Mauro Femminella, Alessandro Parisi, Gianluca Reali

***Abstract*—The goal of this paper is to present a system able to support the network delivery of live events in an expected, future Digital Cinema scenario. This service can consume a large amount of network bandwidth, due to the large volume of transmitted data and to the number of receivers, thus multicast transmission proves to be very useful. Consequently, a key issue of the system is the request routing algorithm, the goal of which is to optimise the QoS-guaranteed delivery of live streams in the backbone, each one towards a set of theatres. We consider the Multi Protocol Label Switching, which has emerged as an elegant solution to meet traffic engineering and resource reservation requirements, and focus on the overall request routing procedure, the mathematical modelling of the problem, and relevant solving algorithms. We present the comparative performance evaluation of these algorithms by means of an extensive simulation campaign performed with the OMNET++ simulation platform.**

***Index Terms*—Digital Cinema, network distribution architecture, multicast, QoS, MPLS, performance evaluation.**

I. Introduction

DIGITAL CINEMA (DC) consists of the replacement of celluloid-based distribution and projection with digital technologies. 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. The transport of DC content (typically movie files of huge size) can be accomplished through either physical media or wired/wireless network distribution, as stated in the Digital Cinema Initiatives LLC (DCI) specifications [2]. The DCI is an entity created by several studios with the primary purpose of establishing uniform specifications for DC. To transfer DC contents to theatres, Distributors may rely on the service provided by a Network Service Provider (NSP).

So far, the service of a DC system is based on the download&play paradigm and DCI does not take into account the projection of live events (e.g., sports and music events), which is, in our view, an application that can bring new interest to theatre-going and thus stimulate the DC business. The IST Project EDCINE (Enhanced Digital CINEma) [1] aims to tackle the issues of distributing live events.

As regards the networking aspects on distributing live events, the data stream delivery to theatres has to be guaranteed in real time. The analysis of a DC survey submitted to European specialists from research institutes and companies [8] showed that end-to-end delay should be below one second, packet loss rate has to be negligible (next to zero) and delay jitter should be in the order of tens of milliseconds.

The high bit rates (tens of Mbps) 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) [7] has emerged as an elegant solution to meet these requirements in IP backbone networks. Moreover, as network bandwidth may also increase due to the number of receivers of a video stream, the use of a multicast transmission is very useful in this scenario in order to deliver live events to multiple theatres contemporarily.

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, 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 simulations performed with a modified version of the OMNET++ simulator [16].

The paper is organised as follows. Section II reports some details about the distribution network. Section III describes the request routing procedure, its mathematical model, and different QoS-enabled multicast routing approaches. Section IV presents a performance comparison among these approaches. Section V reports some concluding remarks.

II. Network Distribution System

Within the framework of EDCINE, we have considered a Content Delivery Network (CDN) able to provide the distribution of pre-recorded films and the delivery of live events to multiple theatres contemporarily. For more details on the EDCINE architecture refer to [3]. Since the delivery of the contents has to be secure and QoS-guaranteed, it is necessary to provide hybrid Virtual Private Network (VPN) links [5]. In the core network, bandwidth provisioning and multicast transmission can be implemented through MPLS [7]. As a matter of fact, MPLS-based VPNs are emerging as the popular choice to build IP VPN due to their scalability, flexibility, cost and the ability to provide IP applications with

Work performed under co-financing by the EU within Project IST-2005-038454 EDCINE (Enhanced Digital CINEma). Authors are with the DIEI, University of Perugia {disorte, femminella, parisi, reali}@diei.unipg.it.

QoS across the network [5][6][7][14]. Moreover, MPLS has been recently updated with the capability of providing QoS-enabled, truly multicast oriented paths [10].

In this scenario, a Provider Edge (PE) node (an IP router or an MPLS-compliant router) connects customers to the provider's backbone network. The theatre site is a local private network, the gateway of which, called Customer Edge, is connected to the PE via a high speed access connection.

As for live events delivery, the CDN routing system is in charge of mapping each live stream with a point-to-multipoint (P2MP) VPN in the MPLS backbone. The multicast tree associated with a live event is clearly dependent on the service requests from exhibitors, the flow ingress point to the backbone and the relevant egress points (associated with the theatres requesting the service). Each live stream has to be routed towards one or more theatres, accounting for service demand, network resource availability, and QoS requirements.

MPLS enables connection-oriented paths (Label Switched Paths, LSPs) to be created within IP-based core networks, de-coupling routing from forwarding functions and providing TE [7]. LSPs are set up before data transmission 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. TE is able to improve QoS by reducing delay and packet losses, while maintaining a high resource utilisation. In order to satisfy the requirements for TE over MPLS, an extension of RSVP signalling protocol (RSVP-TE) has been defined [9], thus allowing the dynamic establishment of LSPs. An LSP can be set up with or without resource reservations. Also, the need to support P2MP services using MPLS-TE LSPs has motivated some enhancements of the MPLS tools [10].

The live streams, captured by cameras in the event site, are preliminarily compressed and assembled in an Outside Broadcast Van. Then, the streams are typically delivered via satellite (this is the "contribution" segment of the network) to the Distributor Head-End, where the data is re-elaborated and finally sent to theatres ("distribution" segment). This scenario brings two limitations to the stream bit rate. The first is the bandwidth availability of the satellite channel (contribution segment) and involves all the transmitted streams; in the DVB-S2 system the limit is about 55 Mb/s [17], whereas in the DVB-S it is 30% lower. The second relies on the particular theatre access network (distribution segment); for instance, the maximum bit rate is 24 Mb/s in the case of ADSL2+. Consequently, the MPEG-4 profiles with 20 Mb/s and 50 Mb/s (2K×1K @ 30 fps) [18] can support DC live events.

III. REQUEST ROUTING PROCEDURE

A. Live Event Management

The NSP in charge of managing DC contents can provide a web server, by means of a theatre administrator can request the access to a live event. From the algorithmic viewpoint, the goal of the NSP administrator is to optimise the QoS-guaranteed delivery of the live streams, setting up properly the P2MP MPLS-TE LSPs. Note that the DC scenario is quite different from a classic QoS multicast problem [13], as a dynamic join or prune of leaves is not expected.

The main steps of the management procedure are:

- the exhibitor books the live event in a time window, in advance with respect to the beginning of the event. For each multicast stream (i.e., live event) the start time, time duration, ingress/egress points of the NSP backbone networks are known;
- the requests from the theatres are collected by an NSP server (Decision Maker, DM), which is also in charge of retrieving the bandwidth availability from monitoring entities;
- the DM runs the multicast QoS routing algorithm to compute the optimal resource allocation in the core network;
- the NSP server sets up the multicast tree for each flow.

B. MPLS-based techniques for QoS-enabled multicast

A P2MP LSP [15][7] has one ingress LSR and one or more egress LSRs, and it is unidirectional. IETF RFC 4875 [11] describes a solution to allow a non-ingress (branch) LSR replicating 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; an ingress LSR uses RSVP-TE as signalling protocol, without requiring any multicast routing protocol in the core network (source routing).

C. Mathematical model

In this subsection, we present a mathematical formulation of our reference QoS-enabled multicast routing problem. The core network topology can be modelled as an oriented graph G=(V,E): V is the set of vertices with $|V|=n$ as cardinality; E is the set of edges with $|E|=m$, numbered from 1 to m, where $e_k=(p,w)$ is the oriented arch from node p to node w. Each multicast flow i is characterised by the triple (s_i, R_i, b_i), i=1,..,L, where s_i is the source node, R_i is the set of receivers with $|R_i|=r_i$, and b_i is the bandwidth amount required by QoS guarantees (see TABLE I).

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

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

whereas the constraints are

$$\sum_{i=1}^{L}Y_{ki}b_i\le B_k,\quad k=1,..,m; \tag{2}$$

$$X_{kp}^{i}\le Y_{ki},\quad i=1,..,L,\ k=1,..,m,\ p=1,..,r_i; \tag{3}$$

$$MX^{i}=A^{i},\quad i=1,..,L\ ; \tag{4}$$

$$X^{i}_{kp}, Y_{ki} \in \{0,1\}, \quad \forall (i,k,p) . \tag{5}$$

In our case, the objective of the problem is twofold: (i) to minimize the amount of allocated network bandwidth; (ii) to maximize the minimum amount of unallocated bandwidth within the backbone, so as to limit potential service blocking for future requests. The positive values α and β represent the relative weights of the two objectives.

As for the constraints of the problem, the first (2) represents a boundary on the amount of bandwidth that can be reserved on each link. The second (3) and third (4) represent the flow conservation constraint for each multicast flow i: a connected tree from s_i to all the r_i receivers (identified by the i-th column of Y) has to be set up. This can be expressed as r_i unicast paths (4), each being a subset of the multicast tree (3) [12].

TABLE I
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×m)	The node-edge incidence matrix of size n×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 i-th flow
	b_i	scalar	Amount of bandwidth required by the i-th flow
	r_i	scalar	Number of receivers associated with the i-th 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× r_i)	Matrix of size n×r_i, whose columns represent the r_i source-destination couples of the i-th 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×L)	Matrix of size m×L, whose columns represent the multicast flows allocation on G, that is Y_{ki}=1 if the edge k belongs to the i-th multicast tree, Y_{ki}=0 otherwise
	X^i	L matrices (m× r_i)	Matrix of size m×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 i-th column of the matrix Y, that is $X^i_{kp} = \begin{cases} 0 & \text{if } Y_{ki}=0 \text{ OR edge } k \notin \text{unicast path connecting} \\ & s_i \text{ to the } p\text{-th receiver of the } r_i \text{ receivers} \\ 1 & \text{if } Y_{ki}=1 \text{ AND edge } k \in \text{unicast path connecting} \\ & s_i \text{ to the } p\text{-th receiver of the } r_i \text{ receivers} \end{cases}$

It is worth noting that the cost function is a non-linear 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 tuple set to be explored in order to find the solution, ranges within $[2^{(mL)(m\min\{r_i\}L)}, 2^{(mL)(m\max\{r_i\}L)}]$.

In the following subsection, we present some alternative algorithmic approaches to find suboptimal solutions to our reference problem. Their performance in terms of both user-oriented and operator-oriented performance figures will be deeply analysed in Section IV.

D. QoS-enabled multicast routing approaches

1) Dijkstra algorithm

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

- Basic Dijkstra (D_noRes) with no bandwidth reservation;
- Dijkstra with bandwidth reservation and Integrity bit set (D_Res_I), where the P2MP LSPs are set-up by RSVP-TE and, in the case of missing resources for a part of a multicast tree, the set-up of the overall P2MP LSP fails [11];
- Dijkstra with bandwidth reservation and Integrity bit unset (D_Res), where the P2MP LSPs are set up by RSVP-TE 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.

2) Dijkstra TE algorithm (D_TE)

We use Dijkstra, adding a preliminary control on the available bandwidth of the links: only the links whose available bandwidth is greater than the flow rate are considered. This information can be made available by OSPF-TE messages [20]. Clearly, each multicast tree computation is influenced by the previously allocated multicast trees. In addition, in order to maximise the minimum amount of unallocated bandwidth within the backbone, we may subtract a bandwidth value equal to *grB* (guard residual Bandwidth) from the speed of each link. Clearly, in this way we set a static, minimum value of the residual bandwidth, thus forcing the algorithm to route streams away from links close to saturation.

3) Original problem solution (LS)

This approach aims at using the commercial optimisation tool LINGO [4] to solve the original problem. Due to the nature of the problem, we expect to find only a local optimum.

4) Synchronous Optimisation (Opt_S)

This approach aims at solving a reduced version of the problem. We name it "synchronous", because it accounts for all service requests simultaneously. Referring to (1), the value of α is set equal to 1, whereas that of β equal to 0 (Steiner tree problem). Thus, the objective function becomes linear and a commercial tool such as LINGO is able to find the global optimum. This approach consists of finding the Steiner tree for all service requests contemporarily [21].

As explained in the previous subsection, in order to control the minimum residual bandwidth the constraint (2) becomes:

$$\sum_{i=1}^{L} Y_{ki} b_i \le B_k - grB, \quad \forall k = 1,..,m . \tag{6}$$

5) Asynchronous Optimisation (Opt_A)

It is the asynchronous version of the previous approach: service requests are considered one by one and each multicast tree computation is influenced by the previously allocated ones. This approach results in using LINGO to compute the Steiner tree for each request every single time.

6) Multicast Near Node First Algorithm (MNF)

Multicast near Node First (MNF) [13] is a directed, Steiner tree computation algorithm. In order to compute the Steiner tree, the MNF approach needs to assign a weight (i.e., a cost) to each link 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. 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, exploiting 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 limits). MNF determines the Steiner tree in an asynchronous way, working on the sub-graph 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 shortest path algorithm.

IV. PERFORMANCE EVALUATION

In this Section, we describe the software platform and the network scenario of the simulation, and finally we present an extensive comparative analysis of the multicast QoS algorithms described in the previous Section.

A. Performance Evaluation Tools

The simulation platform used in the numerical analysis is INET, built upon OMNet++ [16]. Modules able to support RSVP-TE were included into an enhanced INET (E_INET), in order to manage bandwidth guaranteed P2MP LSPs. E_INET sets the optimal multicast trees by using the output files of the optimisation process. The E_INET results are elaborated using Matlab and/or MS Excel. Different optimisation tools have been used: LS, Opt_S and Opt_A were implemented using LINGO 8.0 [4]; MNF was developed in C++; the Dijkstra-based algorithms were implemented as an INET module.

B. Simulated Network Scenario

The simulated network consists of 28 hosts, 3 core routers and 7 access routers. The speed of access links are 2488.32 Mb/s (OC-48), in order 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) for core routers links; 155.52 Mb/s (OC-3) for two specific core links; 622.08 Mb/s (OC-12) for other links.

The multicast data flows are 36. Each flow is CBR (Constant Bit Rate) and is associated with a source (server application), a multicast group (8 client applications), a multicast address, and a bit rate. Each live event is modelled with two different flows, varying the bit rate at application layer (20 Mb/s and 50 Mb/s). Their multicast groups are disjoint. The packet payload size is 1400 B (lower than the Ethernet maximum packet length, 1500 B), and the size of the headers is equal to 39 B (UDP, IP, MPLS, and PPP).

Each P2MP LSP is associated with a logical queue with a buffer size of 20 packets and an amount of guaranteed bandwidth. The multiple queues are managed by the WFQ (Weighted Fair Queuing) discipline [19].

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

C. Numerical Results

The performance figures we have evaluated 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.

We consider three couples of values for α and β (α=1/β) (1):

- α=1 and β=1: the first component of (1), the total network traffic, is dominant with respect to the second component, the Minimum Residual Bandwidth (MRB). We expect the first component to be some orders of magnitude higher than the second. This implies that the target of the original optimisation problem is mainly to minimise Traffic.
- α=0.1 and β=10: this configuration should balance the two contributions, i.e., the original optimisation problem should both minimise Traffic and maximise MRB.
- α=0.01 and β=100: the second component should be dominant, and thus the target is mainly to maximise MRB.

1) Numerical results: user-oriented QoS performance figures

D_noRes, which does not reserve bandwidth, is the only one experiencing packet losses: for a multicast stream, the average packet losses is 5%, whereas the maximum is 79%. Only D_Res_I and D_Res have unreachable receivers, as they do not perform a preliminary check on bandwidth availability. Since the D_Res_I has the Integrity bit set (III.D.1), it shows more unreachable receivers (24.65%) than D_Res (5.9%). The result is that, as expected, the basic Dijkstra computation (not accounting for the available resources) can be used only if the backbone is strongly over-provisioned.

As for the performance in terms of delay parameters, we do not report curves due to space limitations, and present a brief summary of the results below. The maximum and average values of the total server-client delay are obtained; its main component is the queuing delay. D_noRes performs worst since some links prove to be overloaded. D_Res_I and D_Res work very well, but we need to consider that some clients are not served. The D_TE performs slightly better than Opt_S, Opt_A, and LS, which mainly select the outer ring links between the egress nodes in order to minimise the overall hops. In such links, packets experience a higher queuing delay due to a larger amount of traffic. Instead, D_TE minimises the source-destination distances and experiences a lower delay. For D_TE, Opt_S, and Opt_A, the delay proves to be independent of the value of *grB*. Finally, MNF shows a greater delay than D_TE, due to the fact that its main goal is to avoid the critical links rather than minimise the delay or balance the load. We conclude that all the approaches provide end-to-end delay values compliant with the live event delivery service requirements (<1 s).

A parameter linked to the end-to-end delay is the diameter of a multicast tree, defined as the maximum number of links between the ingress-egress couple. As expected, the Dijkstra approaches give the best results due to the nature of the algorithm. D_Res presents a lower mean value due to the pruning mechanism. The others have similar performance.

We define the delay jitter as the standard deviation of the end-to-end delay; maximum and average values are obtained. The results are similar to those of the end-to-end delay analysis. All the approaches provide a delay jitter which is fully compliant with the specifications (tens of ms).

2) Numerical results: operator-oriented performance figures

The values of the overall gross rate (*Traffic*) are shown in Fig.1a. This parameter is related to all the P2MP LSPs links of the multicast streams. For D_noRes, D_Res_I, and D_Res, the P2MP LSPs are identical, since they utilize Dijkstra without checking the available bandwidth. What leads to different *Traffic* values is the resource allocation. The result is that: (i) D_noRes has a value close to 1.13 GB/s, with packets losses due to queue overflow; (ii) D_Res shows a value close to 1.10 GB/s and some P2MP LSPs branches are pruned; (iii) D_Res_I has a slightly lower amount of *Traffic* (0.80 GB/s), since some P2MP LSPs are not completely set up.

The initial bandwidth availability check of D_TE enables all receivers to be reached by the flow. Thus, D_TE has a greater value of total traffic (around 1.2 GB/s) than the previous approaches. MNF performs slightly worse, since it computes the multicast tree as a Steiner tree, trying to avoid potentially critical links: its choice of P2MP LSPs may span further over the network and involve more links. Opt_S and Opt_A focus on total bandwidth minimisation, and thus provide the best performance. Their gain with respect to D_TE and MNF is around 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 for which it also tries to minimise the *Traffic*. For (α,β)=(0.01,100), since the target of LS is mainly to maximise *MRB*, the amount of *Traffic* is higher.

Fig.1b reports the values of *MRB*. It is an indicator of the capability of the network to route new QoS-guaranteed LSPs, thus reducing the probability of service blocking. Simulations show that D_noRes experiences a value close to zero, thus some interfaces are overloaded. For D_Res_I and D_Res the values are low (close to 5 Mb/s). As stated, they also present some unreachable receivers and lower values of *Traffic*. D_TE and Opt_S have similar values, but they are able to deliver a higher amount of *Traffic* (no receivers are blocked). The low *MRB* value of Opt_S is due to the objective functions, which do not account for residual bandwidth. Opt_A shows a better *MRB* (25 Mb/s), as it sets up different P2MP LSPs, leading to larger *Traffic* and *MRB* values. For MNF, *MRB*= 14 Mb/s.

The simulations for D_TE, Opt_S, and Opt_A are repeated with *grB*=50 Mb/s for each link, which ensures enough capacity in core routers to accept at least an additional multicast flow. In this case, Opt_S and Opt_A have very similar values (about 67 Mb/s), even though the P2MP LSPs are not the same. D_TE presents a lower value (56 Mb/s).

LS, for (α,β)=(1,1), not only minimises *Traffic*, but also provides the best *MRB* value (77 Mb/s). This is an expected result, as this configuration, which aims mainly to minimise the handled traffic, also minimises (with lower priority) the *MRB*. Since the other two configurations give a higher importance to *MRB*, we expect at least an equal behaviour, but LINGO is unable to move away from a local optimum.

The total operational cost is defined in accordance with the objective function of the optimisation problem (1) as the weighted sum of the amount of used network resources:

$$Cost = \alpha \times Traffic - \beta \times MRB\,. \tag{7}$$

TABLE II reports the cost values normalised by 1 Mb/s. We consider only those approaches which satisfy all the QoS requirements (Dijkstra TE, Opt_S, Opt_A, MNF and LS). We expect definitely positive cost values for (α,β)=(1,1), close to zero for (0.1,10), and definitely negative for (0.01,100).

All approaches present a computation time compliant with the request routing procedure. Dijkstra algorithms (E_INET embedded) show negligible execution time. The highest computation time is that of LS (1 hour on a standard PC). The use of LS imposes to book live events several hours before the event start, which is a reasonable constraint. However, the computation time can be reduced with a powerful workstation.

As for the RSVP-TE signalling overhead, the values are definitely negligible in all cases, lower than 31.63 KB/s.

D. Discussion

To sum up, Opt_S, which solves a simplified version of the reference problem, presents the best performance: it guarantees the QoS, it minimises the *Traffic*, it is able to control the *MRB* with the *grB* parameter, and it provides the best operational cost with a 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* from zero to the minimum value of link capacity. Then the final choice of *grB* is the one which minimises the operational cost. For instance, for (α,β)=(1,1) the best value of *grB* is 80 Mb/s, for which Opt_S performs better than LS in terms of operational cost, equal to 6733.

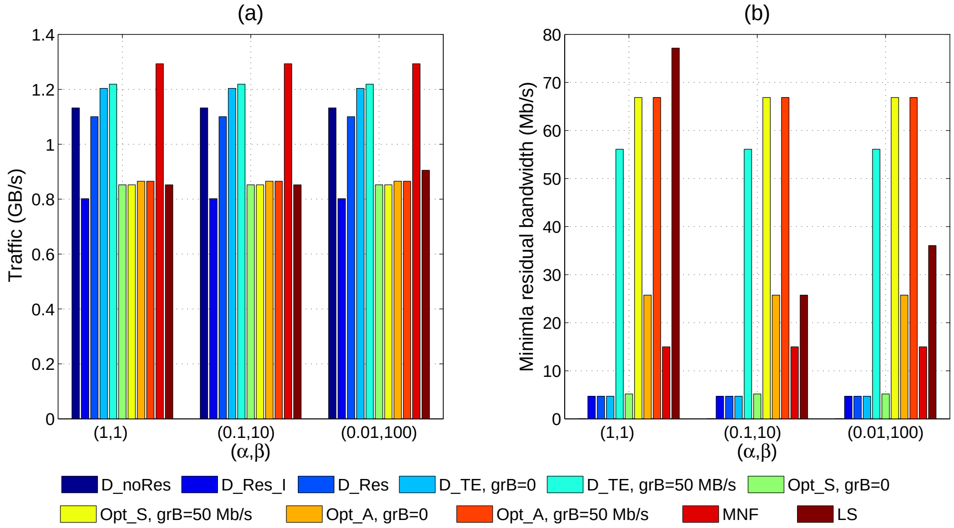

Fig. 1. (a) Total traffic handled by the core network; (b) Minimal residual bandwidth.

TABLE II
OPERATIONAL COST: COMPARISON AMONG THE CANDIDATE APPROACHES.

Approach		**Operational cost**		
		α=1, β=1	α=0.1, β=10	α=0.01, β=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		10329.814	884.620	-1395.152
LS		**6742.000**	424.7	-3511.46

V. CONCLUSION

In this paper, we have presented some key issues related to the DC network distribution, which can indeed be considered as a bandwidth-consuming killer application in the near future. We have especially focussed on the live event delivery within an MPLS backbone and analysed the multicast QoS routing problem associated with the transport of large streams to theatres. We have presented the mathematical model, and then a number of solving approaches, and compared them by means of extensive simulations performed with OMNET++.

We can conclude that, even if the nature of the objective function of the multicast QoS routing problem does not allow the convergence to the global optimum, it is possible to achieve a good solution by using the Opt_S approach. In fact, in the network scenario analysed, Opt_S is able to (i) guarantee the QoS level to support the live event delivery service, (ii) minimise the overall traffic in the core network, (iii) control the amount of *MRB* by varying the *grB* parameter, (iv) provide the best performance in terms of operational cost, and (v) provide a solution with a very low computation time.

REFERENCES

[1] IST EDCINE Project Web Site, http://www.edcine.org/.

[2] Digital Cinema System Specification v1.1, May 2007, http://www.dcimovies.com/press/DCI_Press_Release_3May2007.pdf

[3] D. Di Sorte, M. Femminella, A. Grasselli, G. Reali, "Network distribution of digital cinema contents," IST Mobile and Wireless Communications Summit 2007, Budapest, Hungary, July 2007.

[4] LINGO tool, http://www.lindo.com/products/lingo/lingom.html.

[5] The Virtual Private Network Consortium, http://www.vpnc.org/.

[6] A. Daniel, "IP Virtual Private Networks-A Service Provider Perspective", IEE Proceedings Communications, 151(1), February 2004.

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

[8] EDCINE Project Deliverable D1.1 "Requirements and General Conditions", available at http://www.edcine.org.

[9] D. Awduche et alii, "RSVP-TE: Extensions to RSVP for LSP Tunnels", IETF RFC 3209, December 2001.

[10] S. Yasukawa, "Signaling Requirements for Point-to-Multipoint Traffic-Engineered MPLS Label Switched Paths (LSPs)", IETF RFC 4461, April 2006.

[11] R. Aggarwal, D. Papadimitriou, S. Yasukawa, "Extensions to RSVP-TE for Point-to-Multipoint TE LSPs", IETF RFC 4875, May 2007.

[12] C.A.S. Oliveira, P.M. Pardalos, "A Survey of Combinatorial Optimization Problems in Multicast Routing", Computers and Operations Research, 32(8), August 2005, pp.1953-1981.

[13] M. Kodialam, et alii, "Online Multicast Routing With Bandwidth Guarantees: A New Approach Using Multicast Network Flow", IEEE/ACM Transactions on Networking, 11(4), August 2003, pp. 676.

[14] N.G. Duffield et alii, "A Flexible Model for Resource Management in Virtual Private Networks," SIGCOMM, August–September 1999, USA.

[15] D. Awduche et alii, "Requirements for Traffic Engineering Over MPLS", IETF RFC 2702, September 1999.

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

[17] A. Bertella, et alii, "Laboratory evaluations of DVB-S2 state-of-the-art equipment," EBU Technical Review, January 2007.

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

[19] A. Demers, S. Keshav, and S. Shenker, "Analysis and Simulation of a Fair Queueing Algorithm," SIGCOMM '89, 19(4), September 1989.

[20] D. Katz, K. Kompella, D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2," IETF RFC 3630, September 2003.

[21] B. Wang, et alii, "Multicast Routing and its QoS Extension: Problems, Algorithms, and Protocols," IEEE Network, January/February 2000.