

Autonomic Platform for Synchronous e-Training in Dispersed Organizations

Juan C. Granda · Pelayo Nuño · Daniel F. García ·
Francisco J. Suárez

Received: 12 November 2012 / Revised: 23 September 2013 / Accepted: 27 September 2013 /
Published online: 6 October 2013
© Springer Science+Business Media New York 2013

Abstract Synchronous e-training of human resources uses conferencing services to support training at the workplace. Multimedia data is interchanged between participants in real-time in a synchronous e-training activity providing an enriched learning experience. Large organizations are usually widely dispersed in several sites, so synchronous e-training activities are especially difficult. Most of the sites constitute multicast islands, as IP multicast is available, so the data transport can use IP multicast within the sites of the organization in order to save bandwidth. In this paper, an autonomic synchronous e-training platform based on standard protocols implementing self-management features such as self-deployment, self-organization and self-healing is proposed. The platform deploys an overlay network composed of Real-time Transport Protocol relays acting as multicast/unicast reflectors. The overlay is automatically reorganized according to the joining and leaving of participants and network failures. The use of standard protocols and a modular design enable the interoperability, extensibility and portability of the platform.

Keywords Self-organization · Self-healing · Overlay multicast ·
Interactive multimedia · Session Initiation Protocol

J. C. Granda (✉) · P. Nuño · D. F. García · F. J. Suárez
Department of Computer Science and Engineering, University of Oviedo, Gijón, Spain
e-mail: jcgranda@uniovi.es

P. Nuño
e-mail: nunopelayo@uniovi.es

D. F. García
e-mail: dfgarcia@uniovi.es

F. J. Suárez
e-mail: fjsuarez@uniovi.es

1 Introduction

Synchronous e-training has emerged as a complement to traditional training in large organizations in recent years. Synchronous e-training platforms make intensive use of multimedia conferences involving employees at their workplaces [1].

Features such as audio and video, shared whiteboards, telepointers and collaborative annotation of documents are common in synchronous e-training tools, so a synchronous e-training platform must cope with the multimedia data transport between participants. Thus, a synchronous e-training platform resembles a conferencing platform. However, corporate conferencing usually involves several dispersed groups communicating within conferencing rooms, while users are located at their workplaces in synchronous e-training, so productivity and interactivity between users are improved. Users can send and receive multimedia data to and from all the other users in synchronous e-training activities.

Although recent advances in network technologies have increased the bandwidth of network links, network resources are still considered valuable assets within large organizations. This is especially true in dispersed organizations, as corporate data traverses public networks through virtual private network links and is not limited to the corporate network scope. As a result, traffic from synchronous e-training activities must share the available network resources with other data which is usually crucial for the daily tasks of the organization.

The aim of a synchronous e-training platform is to provide a reliable and highly efficient data transport service with minimum network resource consumption. The latency introduced in data delivery must be as low as possible, so communications can be interactive.

Native IP multicast satisfies all the aforementioned requirements for multipoint group communication [2]. In fact, IP multicast is the best data delivery scenario, as there is no data replication at the network level and it conveys data from sources to receivers with the lowest latency [3]. Network routers must be appropriately configured to support IP multicast. A successful experience in the use of IP multicast supporting very large conferences has been reported in [4]. However, since IP multicast has several security issues and may consume a large amount of network resources [5, 6], the availability of IP multicast is limited, as many Internet service providers do not enable IP multicast in their networks. Therefore, IP multicast is rarely available throughout corporate networks.

The IP multicast service can be emulated using different techniques but at the cost of data transport efficiency. These techniques usually imply the deployment of overlay networks. In addition to an efficient data delivery service, a synchronous e-training platform must be able to automatically configure the data transport overlay network between users. The platform must also adapt itself as network or user failures occur and when users join and leave the activity, exhibiting an autonomic behavior [7].

Finally, platforms in which management issues are not transparent to users are difficult to maintain. If the setup or management of a synchronous e-training activity requires human intervention, highly specialized staff is necessary and the preparation of the activity is a complex and expensive process, as well as being

prone to errors. Furthermore, platforms based on ad hoc solutions not using standard protocols and recommendations are incompatible with other software or hardware conferencing solutions.

In this paper an autonomic platform for synchronous e-training in dispersed organizations based on standard protocols and several multicast/unicast reflectors is presented. The e-training platform implements self-management features such as self-deployment, self-organization and self-healing, providing an efficient and self-managed transport overlay network. The overlay is automatically reorganized whenever a user leaves or joins the activity or a network failure occurs, managing the underlying network resources efficiently to provide real-time communications. Therefore, the platform requires minimum human interaction to configure the overlay and set up activities. Furthermore, the use of standard protocols allows for interoperability with other systems and devices, so the extensibility of the platform is guaranteed.

The remainder of this paper is organized as follows. In Sect. 2, related work on existing techniques for real-time multimedia communication is discussed. The elements that constitute the synchronous e-training platform are described in Sect. 3. In Sect. 4, the design and the operation of the Rendezvous Point are exposed. The self-organization technique used to deploy the transport overlay network is explained in Sect. 5. The synchronous e-training platform is tested in Sect. 6 and the results exposed in Sect. 7. Finally, Sect. 8 contains the concluding remarks.

2 Related Work

Exhaustive analyses of the techniques used to interconnect distributed groups of users by emulating IP multicast services were carried out in [8, 9]. Multicast/unicast reflectors and Application Layer Multicast (ALM) are the most common. Both alternatives are appropriate for synchronous e-training activities, but their efficiency depends on the number of groups to interconnect, the maximum latency allowed or the number of data sources.

2.1 Multicast/Unicast Reflectors

A multicast/unicast reflector is a network entity that forwards incoming multimedia data to one or several destinations, which may be receivers or other reflectors. Thus, it is possible to deploy an overlay network based on reflectors.

One of the initial overlays based on reflectors is described in [10]. The authors propose an architecture enabling unicast users to join multicast groups and participate in communication sessions. An external session controller manages each reflector, determining if incoming users are directly accessible through a multicast network by exchanging Real-time Streaming Protocol (RTSP) messages. Each session controller manages a reflector using the reflector session protocol. This architecture is improved in [11] allowing the session controller to manage several reflectors, although self-healing techniques are not provided.

A technique using autonomous reflectors to maintain application-level connectivity in the case of network failures is developed in [6]. Reflectors decide to migrate, remove, clone or merge with other reflectors in a decentralized self-organized manner, in order to achieve low-cost tree configurations for the distribution of group data. However, each reflector can forward only one media stream (audio, video, chat, etc.).

A web portal to manage conferences and reflectors was developed in [12]. Each reflector includes a mechanism for detecting inactive participants based on timers. An extended version of the reflector is presented in [13], including authentication and authorization features. Self-organizing and self-healing techniques for managing the overlay are not considered.

In the streaming video distribution platform developed in [14] an external server, called Overlay Configuration Management (OCM) server, is used to control the deployment and management of the reflectors. The OCM server receives requests from participants and reflectors, and instructs the participants as to which reflector they should connect to. The decision is made according to criteria such as the round-trip time and the number of participants connected to the reflector. The OCM server uses a TCP connection to communicate with reflectors as a keep-alive mechanism, and is able to relocate participants after a reflector failure. This architecture is built over the RTSP protocol.

The RTSP protocol is combined with the Session Announcement protocol (SAP) to build a live streaming architecture in [15]. Two special entities manage this architecture. First, a SAP/RTSP server announces the availability of new multimedia content and handles client joining requests. Second, an external server, called Resource Manager (RM), builds and manages a three-level overlay topology composed of live media servers, clients and reflectors acting as intermediaries between media servers and clients. The RM decides which relay must serve each client and also reorganizes clients on the fly to improve client quality of service. The management of media servers also relies on the RM.

An architecture to provide multipoint communication in a heterogeneous multicast/unicast environment is proposed in [16]. This is based on open software routers working as H.323 multipoint control units, called S-MCUs, constituting an overlay of S-MCUs. Every S-MCU creates and maintains its own list of participants. They accept joining and leaving requests from participants and contribute to the creation of the overlay. The organization of the overlay is determined by an external server using the information from each S-MCU. This server also manages the session details, including user accounts and permissions.

Finally, a hybrid approach for multi-party videoconferencing supporting Peer-to-Peer (P2P) and reflector forwarding is presented in [17]. Both data forwarding approaches can be used. When using reflector forwarding, some overlay members with multicast capability are able to play the role of a reflector. They forward data from the multicast domain to unicast users and vice versa. They also forward traffic between unicast users. The session management is controlled by an external entity called videoconferencing service center. This manages the session initialization, joining and leaving of overlay members, and connection building between reflectors and unicast users.

2.2 Application Layer Multicast

An ALM is a networking architecture in which the delivery of data between endpoints and management tasks usually rely on endpoints themselves. This allows the emulation of different network services such as IP multicast regardless of the configuration and capabilities of the underlying network. ALM overlays have more flexibility and scalability than overlays based on reflectors, but the latency is increased. Some of them are based on P2P models. The main difference between P2P-based and non P2P-based ALMs is the knowledge about the overlay. Each member propagates its knowledge about the overlay to its peers in P2P-based ALMs. In contrast, a precise knowledge about all the overlay is necessary for proper data distribution in non P2P-based ALMs. The responsibility for maintaining this knowledge may lie in one or more endpoints.

An analysis of self-organizing ALM techniques is developed in [18]. The authors compare distributed and centralized protocols for overlay management. The nodes of the overlay make independent decisions to reorganize the overlay when using a distributed protocol. In contrast, centralized protocols require a representative node responsible for the control of the overlay. This element is usually referred to as the Rendezvous Point (RP) since it also serves as entry point to the overlay. An example of centralized ALM is Application Level Multicast Infrastructure (ALMI) [19]. The RP manages two networks simultaneously. It maintains a control network with a star topology with non-persistent connections, and builds the data delivery network between the participants by estimating the minimum spanning tree. A similar approach is used in [20] for real-time video communications. However, most of the ALMs follow a decentralized approach.

Decentralized ALM overlays, usually based on P2P, achieve higher scalability than centralized overlays. They can be deployed according to several topologies [21], trees [22] and meshes [23] being the most common. These topologies can be combined leading to hybrid topologies, where endpoints can be organized in various local meshes across a global distribution tree in clustered topologies [24], or several distribution trees can be overlapped building a multi-tree topology [25].

P2P-based ALMs can be further classified as structured and unstructured according to whether they are tightly or loosely controlled respectively [26]. In unstructured overlays, a member randomly chooses some members of the overlay as its neighbors, while in structured overlays the neighborhood relationships are established according to specific rules [27]. Some structured P2P-based ALMs allow for deploying overlays where members are organized through hierarchical relations [28, 29]. Unstructured P2P-based ALMs allow for deploying overlays with highly volatile neighborhood relations according to data availability or endpoints contribution [30, 31]. These overlays are mainly aimed at large scale live streaming and video on demand [32, 33]. Finally, some recent ALM overlays follow a mixed approach combining structured and unstructured features [34, 35].

2.3 Analysis and Discussion

Endpoints in an ALM continuously evaluate the performance of their connections to determine the optimal routing paths. They can modify the relations with their neighbors to obtain the most efficient distribution approach, so the scalability of the overlay is usually high. Nevertheless, in spite of this, the latency introduced in communications may be high, as data traverses several hops from sources to receivers (depending on the topology of the ALM), especially in P2P-based ALMs. As a result, ALMs are not convenient for developing real-time activities in large organizations with low latency. On the other hand, a multi-reflector technique is appropriate when the latency of communications has to be low. Minimum latency is achieved when using an overlay network with the reflectors deployed in a full-mesh topology, as data traverses two reflectors maximum.

Many of the solutions based on reflectors are appropriate under the conditions imposed by multimedia real-time activities. However, most of them lack mechanisms for the automatic deployment and initial organization of the overlay, so they must be manually configured by an advanced user before an activity begins. Similarly, when users join and leave an activity, or when a reflector goes down, the overlay is not reorganized by excluding and including reflectors automatically, so the delivery service may become inefficient. Moreover, since the number of distributed groups of users can be assumed to be relatively small, it seems appropriate to use techniques where the management of the reflectors is centralized. The advantage of a centralized management approach is its simplicity and a lower bandwidth consumption than a distributed approach, as the latter usually needs a larger number of control messages between overlay members. Therefore, a technique based on an RP seems appropriate.

Finally, another desirable characteristic is the use of standard protocols and recommendations. Signaling protocols such as the Session Initiation Protocol (SIP), RTSP and H.323, allow for a standard overlay management, so the extensibility and interoperability of the platform is guaranteed. However, RTSP is aimed at large scale live streaming and video on demand. In the case of synchronous e-training activities, protocols such as SIP or H.323 are more suitable, as they are designed for audio and video conferencing. Furthermore, a simple architecture is needed to promote flexibility and portability, so the use of the SIP protocol to manage the overlay is the best choice. The fact that SIP has been adopted as the signaling protocol for third generation mobile telephony also reinforces the suitability of this protocol. Mobile devices can easily be supported by an e-training platform when using SIP.

None of the aforementioned techniques for real-time communications satisfy all these requirements, so a new solution is proposed to be applied in synchronous e-training activities.

3 Architectural Design

The overlay of the proposed synchronous e-training platform is composed of three virtual networks as depicted in Fig. 1. First, a Real-time Transport Protocol (RTP)

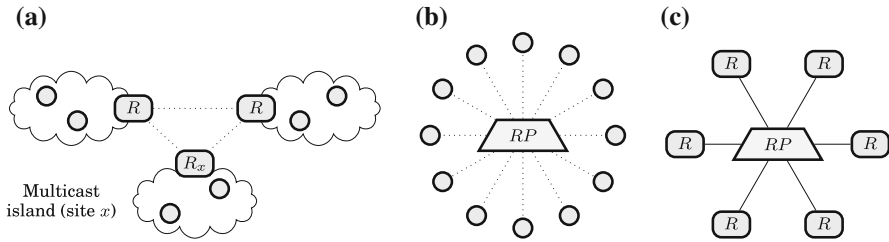


Fig. 1 Virtual networks composing the overlay: **a** relay mesh; **b** signaling network; **c** mesh control network

relay mesh is responsible for delivering multimedia data to participants in synchronous e-training activities. Each RTP relay acts as a multicast/unicast reflector of RTP traffic. Second, a signaling network allows users to participate in activities and facilitates the negotiation of the multimedia configurations of activities. Finally, a mesh control network is used to reorganize the relay mesh according to the joining and leaving of participants and network failures, so the real-time data delivery is efficient. The main advantage of this architectural design is modularity, which allows for developing a flexible and highly portable e-training platform. Thus, the topology and architecture of each virtual network can be changed without affecting the others.

3.1 RTP Relay Mesh

RTP is considered the de facto standard for delivering continuous media such as audio and video through IP networks [36], but it can also be used to transport other time-dependant data. Since multimedia data in a synchronous e-training activity must be delivered in real-time to enable interactive communications, RTP is ideal for synchronous platforms [37]. RTP runs over UDP/IP and separate RTP sessions must be used to deliver each media type. An RTP session represents an association of participants, which is not limited to a specific geographical location or a network scope, involving all the participants in an activity so they can send and receive data to and from the others.

Ideally, IP multicast is available in the underlying network and each RTP session is associated to an IP multicast group, so participants must join various multicast groups in order to receive all the multimedia data. However, the availability of IP multicast in corporate networks is limited. A corporate network can be considered as a group of IP multicast islands interconnected through a non-multicast-capable network. An RTP relay replicates incoming RTP data to participants and other relays. The use of several interconnected relays creates a relay mesh, in which each relay acts as a proxy between the IP multicast island where it is located and the rest of the relays in the mesh. Thus, the relay mesh transparently emulates the behavior of a multicast network to RTP applications despite the capabilities of the underlying network.

A relay combines unicast and multicast delivery, ensuring minimum bandwidth usage. Unicast is used to deliver data to peer relays, whereas IP multicast is used to distribute data to participants within the multicast island. Thus, the forwarding of the

RTP traffic is optimal and guarantees that the number of replicas of each RTP stream is minimal, providing an efficient delivery service [38]. Furthermore, since the relay understands RTP traffic, it is possible to process RTP to perform self-regulation techniques such as transcoding data, merging several RTP streams into one, selecting which stream to forward, etc. Occasionally, a relay may also forward traffic to participants without IP multicast connectivity by sending and receiving multimedia streams directly to and from the participants using unicast. This is useful when a participant joins an activity from outside the corporate network or the relay in the multicast island of the participant is down, so the participant can be redirected to another relay outside its multicast island.

All relays in the mesh are connected in a full-mesh topology, as shown in Fig. 1a, so as to minimize transport delay. Data from a participant must traverse a maximum of two relay hops. Each participant, except those from outside the corporate network, joins an activity from a multicast island where a relay is located, so the participant may be associated with the relay. As a result, participants can communicate with the others sending and receiving multicast streams. The use of a full-mesh topology makes the distribution tree trivial, since relays have to forward multicast data to all peer relays and unicast data coming from the rest of the relays to participants. Traffic must also be forwarded to participants outside their multicast island or connected from outside the corporate network and vice versa using unicast.

In addition to the transport protocol, RTP defines a complementary control protocol, the Real-time Control Transport Protocol (RTCP). All the participants in an RTP session transmit periodic RTCP packets within a variable interval of time which is recalculated according to the joining and leaving of participants to maintain scalability. RTCP provides feedback on the quality of the data delivery and allows participants to maintain a list of all the participants in the RTP session. Therefore, RTCP must also be forwarded by the relay mesh.

When an RTP or RTCP packet arrives, the relay must determine if its destination IP address is a multicast group or the unicast IP address of the relay. If the packet comes from a multicast group, it must be forwarded to all peer relays. Also, the packet must be forwarded to participants outside the multicast island of the relay. If the incoming packet is sent directly to the relay, it implies that the packet comes from a peer relay or a participant outside the multicast island. If the source IP address of the packet does not belong to a peer relay, the packet comes from a participant outside the multicast island and is replicated to all peer relays. In contrast, packets coming from a peer relay are not replicated to the rest of peer relays, but are forwarded to all participants outside the multicast island. A unicast packet, whether it comes from a peer relay or a participant outside the multicast island, is also forwarded to the participants within the multicast island.

3.2 Signaling Network

SIP is a signaling protocol for initiating, managing and terminating multimedia communications in IP networks [39]. Users exchange text-encoded messages to agree on a multimedia configuration. Various extensions have been proposed to the original SIP standard including call control services, presence control services, instant

messaging, mobility and interoperability with other existing telephony systems. The negotiation of the multimedia configuration between SIP entities is supported by the Session Description Protocol (SDP) [40]. Although SIP is mainly applied to Voice over IP (VoIP), which usually involves only two users in the multimedia conference, it can be used to control multipoint conferences. A SIP entity, referred to as the SIP focus, maintains a SIP communication with each participant to implement a conferencing framework for tightly coupled conferences [41].

In the proposed platform, the SIP protocol is used for the management of synchronous e-training activities. An RP plays the role of a SIP focus to implement a tightly coupled conferencing service. Every participant establishes a SIP dialog with the RP while the activity is running. Thus, signaling communications between participants and the RP constitute a virtual network as represented in Fig. 1b.

Participants join an activity using the SIP standard handshake process, which makes the synchronous e-training platform inter-operable with most SIP applications and devices, such as VoIP phones. Standard authentication mechanisms can also be used with SIP. Moreover, the RP implements a SIP registrar server, so participants may register to upload their current locations. Therefore, it is possible to implement dial-out scenarios for joining an activity, where the RP invites a registered user to participate in the activity.

3.3 Mesh Control Network

The mesh control network is a virtual network established between the RP and the RTP relays during synchronous e-training activities. The RP uses TCP connections with the relays to notify changes in the topology of the relay mesh. These changes are mainly due to inclusions and exclusions of relays from the mesh as participants join and leave or when network failures occur. Figure 1c shows the connections established between the RP and the relays.

Relays must contact the RP as soon as possible to register with it, so the RP has an updated list of all the relays. It is not necessary that all the relays participate in an ongoing activity. In fact, a relay will not be part of the relay mesh if there is no participant in its multicast island. When an activity begins, the RP disseminates the organization of the relay mesh to all the relays involved in the activity. As a consequence, a relay may be in three states: unregistered, when it is not registered with the RP; registered, when it has registered with the RP; and active, when it is part of the relay mesh of an ongoing activity.

3.4 Theoretical Model of Traffic

Participants in a synchronous e-training activity interchange multiple types of data such as audio, video, annotations on a shared whiteboard, etc. Each type of data is delivered using its own RTP session. It is possible to predict the number of RTP streams that traverse each relay in the platform as a function of the number of relays and the number of participants in the mesh. An analytic model of the number of streams that are received and forwarded by each relay in the mesh for an RTP session is proposed in [38]. If the streams have a constant bitrate or an upper bound

for their bitrate can be calculated, as occurs with audio and video streams, the equivalent network bandwidth consumption can be computed.

The model assumes some simplifications. First, participants are supposed to send a constant number of streams to the RTP session, which is quite usual for audio and video. Second, the relay uses IP multicast to communicate with multicast-capable participants, while it uses unicast to communicate with peer relays and participants outside the IP multicast scope. Third, signaling and control traffic are excluded from the model. Finally, the model predicts the traffic for a single RTP session. If several types of data are used in an e-training activity, the model should be applied to each session individually.

A relay is deployed at each site of an organization (R relays). Let S_i be the number of participants that send data to relay i , which is responsible for forwarding the streams to the rest of the RTP session. Let U_i be the number of participants communicating with the relay using unicast. These are further divided into participants that send and receive data (US_i) and participants that only receive data (UR_i):

$$U_i = UR_i + US_i$$

Since a relay receives all the streams of the RTP session, the number of incoming streams to the relay is defined as:

$$\sum_{j=1}^R S_j \quad (1)$$

The streams forwarded by relay i can be divided into three groups. First, the relay forwards incoming traffic to peer relays. These streams are delivered using unicast transport:

$$S_i \times (R - 1) \quad (2)$$

Next, the number of streams that the relay forwards to the unicast participants is defined as:

$$\sum_{j=1}^R (S_j \times U_i) - US_i \quad (3)$$

The third group is made up of the streams coming from peer relays and the unicast participants, which the relay forwards to the multicast group:

$$US_i + \sum_{j=1, j \neq i}^R S_j \quad (4)$$

Thus, the number of outgoing streams from the relay is the result of (2) + (3) + (4).

4 Rendezvous Point

The RP is the essential element for the deployment and management of the synchronous e-training platform. The RP manages the automatic organization of the

communication mesh between relays modifying the mesh topology according to changes caused by the joining and leaving of participants. The RP is also responsible for recovering and reorganizing the relay mesh when a relay in the mesh goes down. If a relay goes down, the RP must redirect the participants in the multicast island of the relay to other relays, so they can continue sending and receiving multimedia data. To do so, the RP must collect information about relays and their associated participants in order to maintain the integrity of the relay mesh.

The RP is the entry point to the synchronous e-training platform. All relays must register with the RP before the beginning of a synchronous e-training activity. Registered relays may be included in the relay mesh of future or ongoing activities. Participants may optionally register to the RP using SIP REGISTER messages, making their locations public to the RP.

Registration requests from a relay contain information about the relay and its capabilities to support activities. The registration of the relay with the RP has an expiration time to provide some feedback concerning the availability of the relay. The relay must re-register with the RP when the registration expires. Once the RP starts an activity, the relay may be requested to participate in the relay mesh of the activity. Nevertheless, there may be registered relays not participating in the activity if there is no participant joining from the site where the relay is located.

Participants send a SIP INVITE to the RP to join an activity. The RP authenticates users and provides the multimedia configuration of the activity. This configuration depends on the site where the requesting participant is located. The participant is associated with the relay located in the same multicast island, so it uses IP multicast to communicate with the relay.

A participant outside a multicast island may also join an activity. The RP associates the participant with a relay according to the capabilities of the relays in the mesh. This requires additional resources at the relay to forward traffic to this participant, as the participant will communicate with the relay using unicast. In some cases, the RP may reject the joining request from a participant outside the corporate network or exclude participants from the e-training activity when their relays go down.

However, the centralized architectural design for the control of the overlay has some drawbacks. All the overlay relies on the RP, becoming a critical point of failure. The consequence of a failure is only one of the criteria that can be used to analyze the different service failure modes originated by a disruption [42]. Consequences affecting the availability, safety, confidentiality or integrity of the overlay may be classified as fully-disruptive (FD), semi-disruptive (SD) and non-disruptive (ND) [43]. Consequences can be further divided into operational and functional consequences. Operational consequences are those involving the overlay network as a whole entity, preventing participants from developing e-training activities. On the other hand, functional consequences are those affecting only the quality of experience of participants in the e-training activities.

A failure or a network disruption affecting the RP implies different functional and operational consequences according to the activity state. During an ongoing activity, an RP failure does not imply fully-disruptive functional consequences. The RP is not responsible for delivering data, so the failure does not prevent participants

from interchanging data. Nevertheless, the self-organization and self-healing techniques cannot be triggered when a participant leaves the activity or when a relay fails. This may lead to a waste of resources, as data streams may be unnecessarily forwarded to relays with no participants. Therefore, RP failures in an ongoing activity lead to a semi-disruptive operational state.

An RP failure also implies that the availability of the overlay decreases. New participants cannot establish a SIP dialog with the RP, so they cannot join the e-training activity. Thus, an RP failure in a partially established activity prevents participants from taking part in it, leading to a fully-disruptive operational/functional state.

To overcome this scenario, fault-tolerant hardware solutions like mirroring and hot spare configurations can be applied to RP. Since the RP is unique in the platform, a solution based on mirroring or redundant hardware is feasible. This is commonly used in 24×7 bootstrap servers of many P2P overlay networks.

E-training activities can be carried out with minimum costs. The costs of the platform can be measured using several metrics: economical, resource consumption and management. Firstly, the economic cost of the platform comes from the equipment required for the RP and the relays. A relay can be run in a regular PC, while the RP requires fault-tolerant hardware to guarantee the resilience of the platform. Only one relay and the RP are required for small scale activities when using a star topology for the relay mesh. Secondly, the resources consumed by the platform depend greatly on the dispersion of participants as analyzed in Sect. 7. Finally, the self-management characteristics of the platform help to reduce the management costs and the requirements of specialized staff.

Moreover, security is critical in corporate synchronous e-training. Many issues may affect the confidentiality, integrity and availability of the platform as pointed out in [44]. The platform allows for the use of the Secure Real-time Protocol (SRTP) for the delivery of data to provide data confidentiality and integrity. However, various security issues remain open such as source authentication in multicast multimedia communications over SRTP with multiple sources. A participant may impersonate other participants. This issue can be alleviated using access control lists in the relays, so data is only forwarded from authenticated participants.

5 Self-deployment and Self-organization Techniques

This section describes the technique for the autonomic deployment and management of the overlay. Our proposal is tailored to geographically dispersed organizations and assumes that an RTP relay is located at each site of the organization and native IP multicast is available at each site. The technique consists of several stages that take place chronologically, but which can be interlaced as participants join and leave an activity or when relay or network failures occur. The technique can be divided into the following stages:

- Registration of relays.
- Joining and leaving of participants.

- Deployment and management of the relay mesh.

5.1 Registration of Relays

Once an RTP relay has started running, it establishes a temporary TCP connection with the registration service to register with the RP. The relay provides the RP with an XML document, containing the following information:

- The IP address and port for establishing the control channel between the relay and the RP.
- The IP unicast addresses and port range available for establishing the data channels between the relay and its peer relays in the mesh or participants outside its multicast island.
- The IP multicast addresses and port range for establishing the data channels between the relay and the participants within its multicast island.
- The identifier of the site where the relay is located.
- The maximum number of participants supported by the relay using IP multicast.
- The maximum number of participants supported by the relay using unicast.

When the RP processes a registration request correctly, it responds to the relay, indicating a successful registration. Then, the TCP connection is closed. The registration has an expiration time, so the relay must re-register with the RP periodically. Hereafter, the relay waits for the RP to reestablish the TCP connection. This means that the RP is requesting the relay to participate in an ongoing activity. In this case, the relay becomes active and receives an XML document from the RP with information about ongoing activities and its peer relays in the relay mesh. The TCP connection remains open while the activity is running.

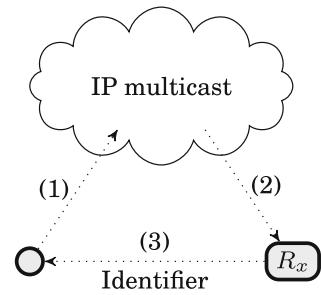
5.2 Joining and Leaving of Participants

A participant must contact the SIP focus module of the RP in order to join a synchronous e-training activity. If the RP accepts the joining request, it associates the participant with a relay in the relay mesh. This relay may have been part of the mesh previously, or the RP may have included the relay in the relay mesh as a result of the participant joining. In either case, the participant must obtain a site identifier, so the RP can associate the participant with the relay.

Two steps are identified when a participant joins an activity. Before joining an activity, the participant must be informed of the multicast island in which it is located by the corresponding relay. Then, the participant can join the activity by establishing a SIP dialog with the RP, indicating the identifier of the multicast island.

As shown in Fig. 2, each relay has an identification service bound to a multicast address that is known by all participants. Participants must send a multicast message to request information from the identification service. When a relay receives an identification request, it responds to the participant with a message indicating the identifier of the site where the relay is located. Note that the relay does not store any

Fig. 2 Identification of the multicast island



information about participants, as it merely responds to all identification requests received.

Once a participant has learned the site from which it will take part in the activity, it must contact the RP. The RP plays the role of a SIP focus, as all participants must establish a SIP dialog with the RP to join an activity. The participant sends a SIP INVITE message to the RP in order to join the ongoing activity. The SIP message contains the SIP Uniform Resource Identifier (URI) of the activity, the SIP URI of the participant and the identifier of its multicast island. The identifier of the multicast island is carried in an optional *Organization* header within the INVITE message.

The RP associates all the active relays with the participants that have joined the activity from the multicast island where the relay is located. When the RP accepts a new participant, it answers with a SIP OK message containing the description of the media that is transmitted during the activity. IP multicast is used to transport data between the participant and its local relay.

If there is no relay available at the multicast island from which a participant has requested participation in the activity, or if the identification process has failed, the participant does not include an *Organization* header in the INVITE message. In this case, the RP associates the participant with the relay with the least number of participants served and the participant uses unicast to send and receive data to and from the relay. The unicast address of the relay is specified in the SDP description. Nevertheless, if the remaining relays are assigned an excessively large number of participants, the RP will deny the entrance of a participant to the activity.

A participant leaves the activity by sending a SIP BYE message. The RP may also exclude a participant from the activity with a SIP BYE message. In the case of a participant leaving the activity ungracefully, the relay associated with the participant detects that the participant has left the activity, as the relay snoops RTP traffic and a timer is used to timeout participants. Thus, the relay can notify the RP of participants leaving the activity.

5.3 Deployment and Management of the Relay Mesh

As previously mentioned, the RP maintains a list of the registered relays, associating them with the identifier of the multicast island in which they are located. A relay is active if at least one participant joins a synchronous e-training activity from the

multicast island where the relay is located. Thus, the relay mesh only involves active relays. Next, the two processes involved in the management of the relay mesh are described: the inclusion and the exclusion of relays.

Figure 3 illustrates the inclusion of a relay in the relay mesh. The relay is included when it becomes active. Figure 3a shows a situation with four registered relays with the RP. A new participant from multicast island *a* establishes a SIP dialog with the RP to join the ongoing activity. Therefore, as shown in Fig. 3b, relay R_a becomes active and all relays must be informed of the new mesh organization. The relay mesh is established among relays R_a , R_c and R_d . The RP sends an inclusion message to notify relays R_c and R_d that a new relay has been included in the relay mesh. When relays R_c and R_d receive the inclusion message, they insert the peer relay included in the message into their redistribution lists. The RP also sends an inclusion message to relay R_a with information about all the active relays in the mesh. The inclusion messages contain information such as the IP address and a range of ports of one or several peer relays. The RP generates inclusion messages according to the information provided by relays during the registration process. Figure 3c shows the relay mesh deployed among multicast islands *a*, *c* and *d* after processing the inclusion messages.

Figure 4 illustrates the process carried out to exclude a relay from the relay mesh. A relay is excluded when it becomes inactive. The rest of the relays must be instructed to stop forwarding data to the inactive relay. In Fig. 4a, the participant from multicast island *d* leaves the activity. As a result, the RP sends an exclusion message to relay R_d , as shown in Fig. 4b. The relay processes the exclusion message and disconnects from the relay mesh. In addition, a removal message is sent by the RP to all active relays (R_a and R_c) to notify that relay R_d has become inactive. The final organization of the relay mesh is shown in Fig. 4c.

The RP is responsible for identifying the cases in which the joining or leaving of a participant in the activity implies a modification of the organization of the relay mesh. This ensures a minimum number of RTP streams forwarded between multicast islands, since those with no participants in activities are excluded.

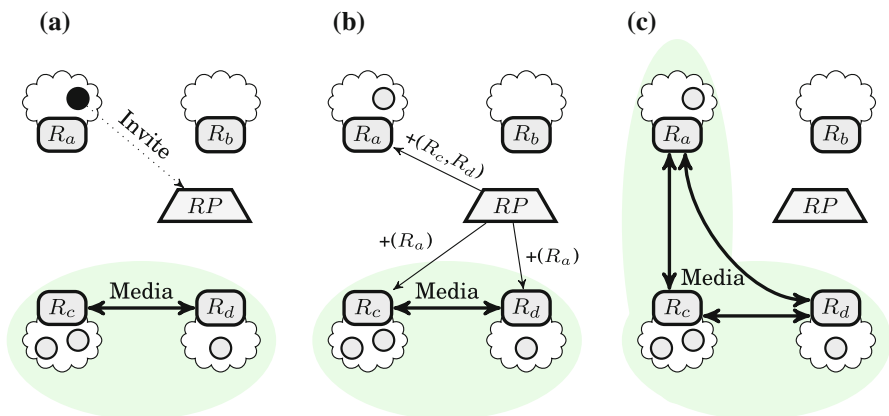


Fig. 3 Inclusion of a relay in the relay mesh of a synchronous e-training activity

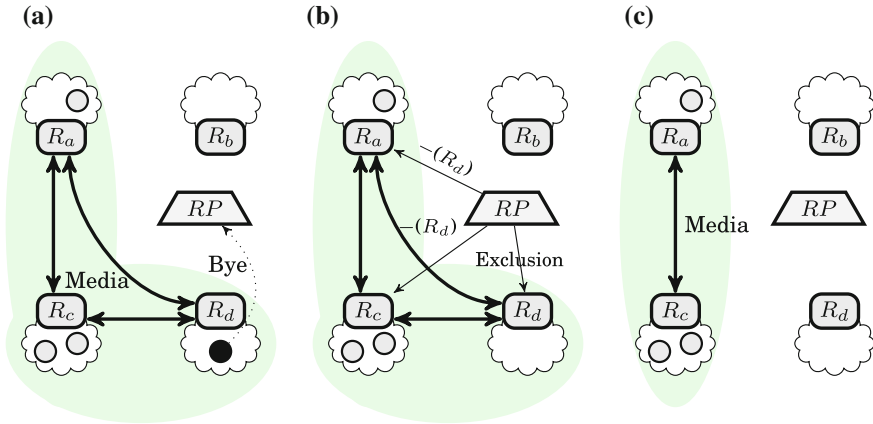


Fig. 4 Exclusion of a relay from the relay mesh of a synchronous e-training activity

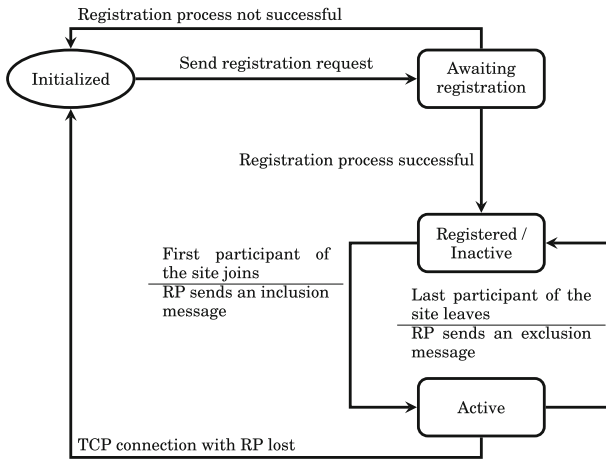


Fig. 5 State diagram of an RTP relay

Figure 5 details the states and transitions between states of a relay as a consequence of the messages received from the RP. The first state is *Initialized*. In this state the relay is running, but it is still not registered with the RP. Next, the relay sends the registration XML message to the RP and its state changes to *Awaiting Registration*. When the RP successfully processes the XML and the relay receives the confirmation, the relay shifts to *Registered* state. If the registration process is not successful, the relay returns to the *Initialized* state. Once a relay is registered, it starts the identification service.

When the first participant of the multicast island of the relay is accepted by the RP, the RP sends an inclusion message to the relay with information about the current organization of the relay mesh. Once the inclusion message is processed, the relay becomes *Active*, forwarding traffic to peer relays and participants.

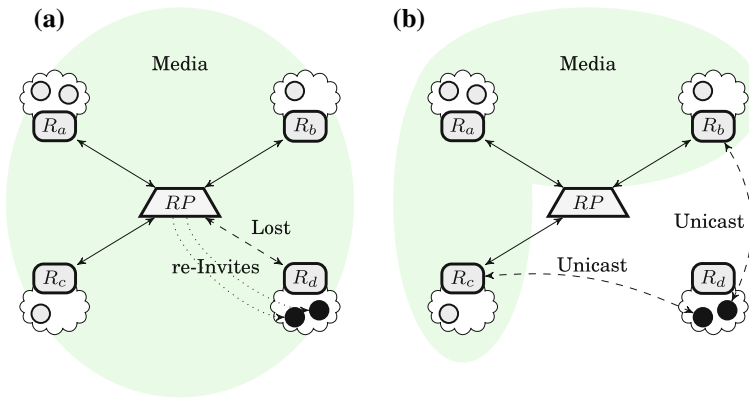


Fig. 6 Self-healing technique

A relay in *Active* state moves to *Registered* state when it becomes inactive, that is, the last participant supported by the relay leaves the ongoing activity. The relay receives an exclusion message from the RP indicating this situation.

The RP keeps an open TCP connection with each active relay. This TCP connection is used as a keep-alive mechanism. The RP eliminates a relay from the relay mesh when its TCP connection is lost. In this case, the relay moves from *Active* state to *Initialized* and it must repeat the registration process.

5.4 Self-healing Technique

A self-healing technique has been implemented in the platform that allows the continuity of the service in spite of relay failures. Whenever a relay goes down, the RP associates the participants of the relay to another active relay in the relay mesh. In this way, the participants still send and receive traffic to and from ongoing activities.

As presented above, the RP keeps an open TCP connection with each active relay and maintains updated information about the identity of participants in each relay. Therefore, when a TCP connection is lost, the RP knows the participants that must be redirected to another relay. The redirection process is illustrated in Fig. 6. Figure 6a shows an activity with four active relays. When the TCP connection between the RP and relay R_d is lost, the RP sends SIP re-INVITE messages to the two participants of relay R_d to change their configuration for data delivery. These SIP messages contain SDP descriptions which redirect traffic from the participants to other relays using unicast. Specifically, a participant is redirected to relay R_c and the other to relay R_b . The final organization of the relay mesh is shown in Fig. 6b.

When the relay is up again a self-optimization technique takes place. The relay registers with the RP and the redirection process is undone. The RP maintains a list with the changes made in the relay mesh, so it is able to roll back any of them.

The redirection process has some influence on the performance of the overlay. Although the number of incoming RTP streams remains the same for all the relays, the number of outgoing RTP streams increases in those relays to which participants are redirected. Let U_{ri} be the number of redirected participants to relay i . These participants can further be divided into participants sending and receiving data (US_{ri}) and participants that only receive data (UR_{ri}). The number of streams coming from the peer relays of relay i decreases by US_{ri} streams, but these streams are now received directly from unicast participants. On the other hand, the additional outgoing streams can be calculated as:

$$US_{ri} \times [(R - 1) + (U_{ri} - 1) + U_i + 1] \quad (5)$$

The streams generated by the redirected participants (US_{ri}) must be forwarded to the peer relays, the redirected participants, the rest of unicast participants and the multicast island of the relay. Taking into account the definition of redirected participants ($U_{ri} = US_{ri} + UR_{ri}$), Eq. (5) can be expressed as follows:

$$US_{ri}^2 + US_{ri} \times (UR_{ri} + U_i + R - 1) \quad (6)$$

Thus, the upstream bandwidth consumption at the relay increases exponentially with the number of redirected participants sending data.

6 Experimentation

The e-training platform has been completely implemented and is fully functional. It has been successfully used to support real synchronous e-training activities within a steel making company [37]. However, the assessment of the performance, robustness and resilience of the overlay requires exhaustive tests that cannot be carried out in a production environment. A model has been developed using the ns-3 simulator to perform the tests. The model simulates the operation of the network of a geographically dispersed organization and all the entities of the platform, including participants, relays and the RP.

6.1 Network Model

The network model simulates the network of an organization dispersed in multiple sites where IP multicast is available, and there is a relay in each site. An ideal Wide Area Network (WAN) connects every site with the RP through 20 Mbps network links with a delay of 20 ms. The data rate of the network link connecting the RP to the WAN is 10 Mbps and introduces no delay in communications.

The range of possible overlay deployments is extremely high. Many parameters such as the number of relays in the mesh, the number of participants, the balance of participants between relays and the number of unicast participants can be varied, resulting in different throughput results. Only one parameter is varied at a time in the tests in order to evaluate the effect of each parameter individually.

Moreover, it must be noted that the full-mesh deployment of the overlay might make the relay mesh incur in scalability issues for a high number of

participants, since data replication grows with the number of relays and participants. Nevertheless, real-time interactive communication services usually involve small groups of participants, so simulations with a few dozen participants are representative of the environments in which the platform can be used. A complete assessment of the scalability of the relay mesh is carried out in [38].

6.2 Participant Modeling

All the participants are modeled in order to simulate the traffic generated during the synchronous e-training activities. A participant can use several kinds of media types to communicate with other participants in an activity. Typically, audio and video are the most commonly used media types, but other media types such as shared whiteboard annotations and telepointers can also be used. Thus, a participant is represented as an entity that generates audio and video streams, annotations in the shared whiteboard and telepointer movements. All this information must be forwarded to the rest of the participants.

A participant waits before joining the activity. The waiting time is simulated using an exponential random variable $Exp(\lambda_w)$. The participant establishes a SIP dialog with the RP in order to join the activity and negotiate the media configuration, and closes the dialog when leaving gracefully. Once the participant has joined the activity, he or she remains joined until the end or leaves unexpectedly without notifying the RP with a preset probability p_l .

The audio and video streams of a participant are activated regularly. A video stream is a simulated 160×120 variable bitrate H.264 video stream at 10 frames per second. An audio stream is a simulated constant bitrate iLBC audio stream with a packetization time of 20 ms. Both the interval between activations and the duration of the streams are assumed to be normal random variables. Thus, a participant generates an audio stream of duration $\mathcal{N}(\mu_{al}, \sigma_{al}^2)$ after an elapsed time $\mathcal{N}(\mu_a, \sigma_a^2)$, and generates a video stream of duration $\mathcal{N}(\mu_{vl}, \sigma_{vl}^2)$ after an elapsed time $\mathcal{N}(\mu_v, \sigma_v^2)$.

The format of the annotations in the shared whiteboard and the telepointers used by participants are described in [37]. A participant generates an RTP stream containing their annotations in the shared whiteboard, while another RTP stream is used to convey the information of the telepointer. Similarly to audio and video, the activation time and the duration of these streams are assumed to be normal random variables. Thus, a participant generates an annotation stream of duration $\mathcal{N}(\mu_{wl}, \sigma_{wl}^2)$ after an elapsed time $\mathcal{N}(\mu_w, \sigma_w^2)$, and generates a telepointer stream of duration $\mathcal{N}(\mu_{tl}, \sigma_{tl}^2)$ after an elapsed time $\mathcal{N}(\mu_t, \sigma_t^2)$.

Obviously, real activities are usually moderated in order to manage the interactions among participants and avoid excessive network resource consumption due to many multimedia streams. Floor control protocols can be used to share the data channels between the participants of an activity. The maximum number of data streams that can be activated simultaneously depends on the desirable interactivity for the activity. Since this number is difficult to establish as floor control policies depend greatly on the kind of activity to be carried out, three scenarios are considered in the tests: low, medium and high interaction. In a low interactive

activity only 2 audio streams and 2 video streams are allowed simultaneously. In a medium interactive activity these numbers increase to 4 audio streams and 4 video streams. Finally, 4 audio streams and 6 video streams are allowed in a highly interactive activity. The maximum number of simultaneous annotation and telepointer streams are 2 regardless of the type of activity.

Two roles can be identified in the participants of a synchronous e-training activity: one instructor and many regular participants. The instructor is a participant who uses all the media types continuously to emulate the speech of the instructor during the activity. The rest of the participants issue floor requests to use the data channels. This situation closely resembles the behavior of users in synchronous e-training activities where participants usually interrupt the activity to ask questions and the instructor can be seen and heard throughout the duration of the activity. In all cases, the requests from participants to use the data channels are granted in a first-in first-out order.

7 Results

Simulations were conducted to assess the performance and the self-healing ability of the overlay. Table 1 shows the settings used during the simulations. Each test simulated a balanced synchronous e-training activity of 1 hour where each relay serves the same number of participants and was repeated 7 times.

Figure 7 compares the average bandwidth consumption in the network links connecting the sites to the WAN depending on the floor control policy used. Figure 7a shows the bandwidth used in an activity with 100 participants when increasing their dispersion (the number of relays in the mesh), while the number of participants is varied for an activity with 10 relays in Fig. 7b. As can be seen, the traffic rapidly grows with the dispersion of participants, but the growth becomes asymptotic due to floor control policies. However, although the average bandwidth consumption is limited, some relays may consume significant bandwidth when senders are co-located in the same sites. To alleviate this issue, floor control decisions can be taken to balance senders between sites.

Table 1 Simulation settings

Waiting time to join (s)	λ_w	1/45
Unexpected leaving probability	p_l	0.05
Audio activation interval (min)	(μ_a, σ_a)	(25,3.3)
Audio stream duration (s)	(μ_{al}, σ_{al})	(15,2.6)
Video activation interval (min)	(μ_v, σ_v)	(40,8.2)
Video stream duration (s)	(μ_{vl}, σ_{vl})	(30,9.5)
Annotation activation interval (min)	(μ_w, σ_w)	(16.6,1.3)
Annotation stream duration (s)	(μ_{wl}, σ_{wl})	(60,10)
Telepointer activation interval (min)	(μ_t, σ_t)	(20,1.6)
Telepointer stream duration (s)	(μ_{tl}, σ_{tl})	(15,2.5)

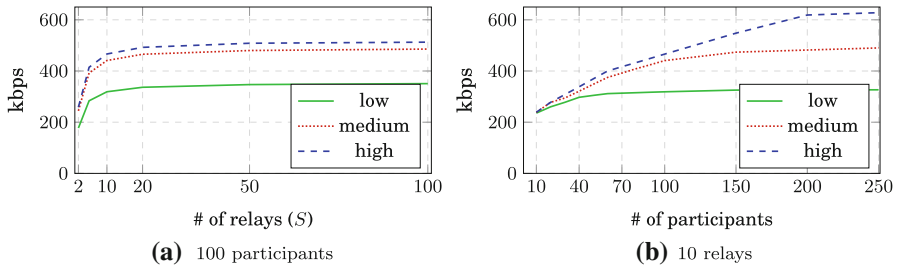


Fig. 7 Average network bandwidth consumption in the network links of the sites depending on the interactivity of the activity

The traffic also grows asymptotically with the number of participants, although with a lower pace as depicted in Fig. 7b. For activities with a low number of participants the results are similar for all the interactivity scenarios, since not all the floors are granted simultaneously in higher interactive scenarios and a similar number of data streams are generated by participants. The small differences in the traffic observed between the three interactivity scenarios are due to control messages. The RTP session bandwidth used to compute RTCP transmission intervals depends on the maximum number of simultaneous floor holders, so a high interactivity level also implies additional RTCP traffic. However, when increasing the number of participants, the bandwidth used in the three interactivity scenarios diverges as more floors are held simultaneously in higher interactivity scenarios and more data streams flow throughout the platform.

The bandwidth consumption for the medium interactivity scenario is broken down in the stack area chart of Fig. 8. Figure 8a shows that video is clearly the most resource-consuming media type, while the average bandwidth consumed by annotations and telepointers is negligible compared to audio and video. On the other hand, two regions can be identified for each media type in Fig. 8b. The bandwidth consumption steadily grows with the number of participants initially when the average number of floors granted is less than the maximum. Once the maximum number of floors granted is reached, the bandwidth consumption increases asymptotically due to control messages from new participants.

The bandwidth consumption for the medium interactivity scenario is also compared to a delivery service based on unicast with the same floor control policy

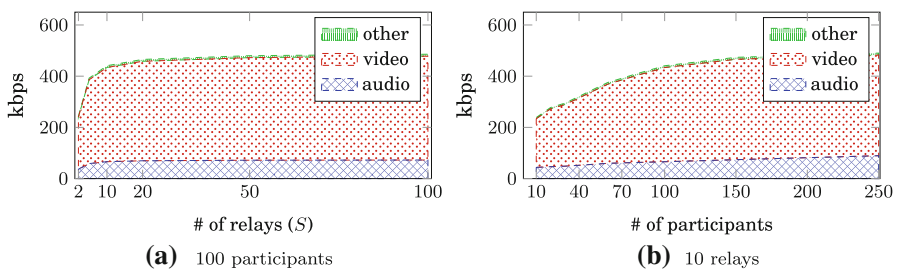


Fig. 8 Average network bandwidth consumption in the network links of the sites for each media type

Fig. 9 Average network bandwidth consumption in the network links of the sites when using unicast and the relay mesh

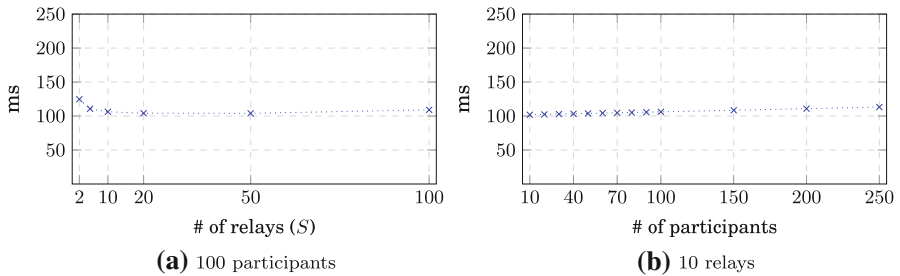
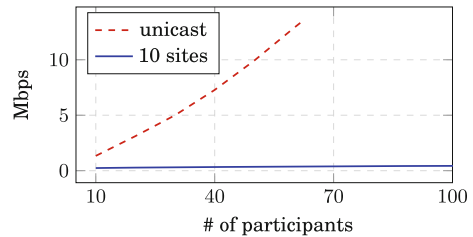


Fig. 10 Time required for the self-healing technique

in Fig. 9. The unicast traffic grows exponentially with the number of participants in spite of using floor control compared to the slight growth when using the relay mesh.

Figure 10 depicts the time required for applying the self-healing technique, which is triggered when an active relay goes down so the participants served by the relay are redirected. This encompasses the detection of the failure, the modification of the relay mesh, the selection of a new relay for each participant, and the subsequent SIP dialogs between the RP and the participants. Figure 10a plots this time for an activity with 100 participants and varying the number of relays in the overlay, while Fig. 10b plots the time for an activity with 10 relays and a different number of participants. The time required to stabilize the overlay after a relay failure rapidly decreases with dispersion for activities with a low number of relays as seen in Fig. 10a, since fewer participants have to be redirected when increasing the number of relays. However, this time increases for highly dispersed activities, as many relays have to be informed when reconfiguring the relay mesh. On the other hand, the stabilization time is directly proportional to the number of participants in the activity as shown in Fig. 10b.

The failure of a relay has some impact on user experience. The failure prevents participants of the site of the relay from receiving and sending data from and to the rest of the activity until the self-healing technique is completed successfully, which leads to a burst of lost packets. Lost packets in continuous media streams produce disturbances, such as audio glitches and video artifacts, which decrease the quality perceived by users. Thus, the quality impact can be measured by means of the number of consecutive lost packets during the re-direction process. On the other hand, packet loss is unacceptable in non-continuous media streams such as annotations, so forward error correction techniques must be used. In this case, the amount of extra information in data packets depends on the maximum packet loss burst to cope with.

Some quality decrease of the video and telepointer streams can be assumed, since video streams only convey the participants’ talking heads and interpolation techniques can be used to obtain smooth movements of the telepointers on the shared whiteboard. Conversely audio streams are critical, especially that coming from the instructor, so the impact of packet loss on quality may be significant. This impact depends on whether the packets lost contain silence or data from talk spurts.

Figure 11 shows the average packet loss observed in the received audio streams when applying the self-healing technique. The average number of consecutive lost packets of the streams affected by packet loss is depicted by a blue dotted line. These streams are those received at or sent from the site of the failing relay. The average number of consecutive lost packets observed in all the streams is depicted by a solid red line. Figure 11a plots packet loss varying the number of relays in the overlay, while Fig. 11b plots packet loss as a function of the number of participants.

Participants are likely to still receive most of the data streams through IP multicast in activities with a low number of relays when a relay failure occurs, since many participants are co-located. These streams are not affected by packet loss. In the case of the streams with packet loss, the average number of consecutive lost packets slightly increases with dispersion, as the time required to stabilize the relay mesh after a relay failure also increases with dispersion. However, when the dispersion of participants grows, most of the streams received by participants come from the relay mesh, so the likelihood that a stream comes from the site of a failing relay is higher, so packet loss increases. When increasing the number of participants, the number of streams flowing throughout the platform also increases, so the failure of a relay has less influence on the total average of packets lost per stream.

Finally, the time for applying the self-optimization technique is also analyzed in Fig. 12. This is triggered when the failing relay is up again to undo the redirection of participants. The time is plotted as a function of the number of relays in the overlay in Fig. 12a, and the number of participants in Fig. 12b. This time shows a similar trend to the time required for the self-healing technique and depends on the dispersion and number of participants. However, the former is significantly higher than the time required to apply the self-healing technique, as it includes the time required for the registration of the relay with the RP before the redirection process can be undone.

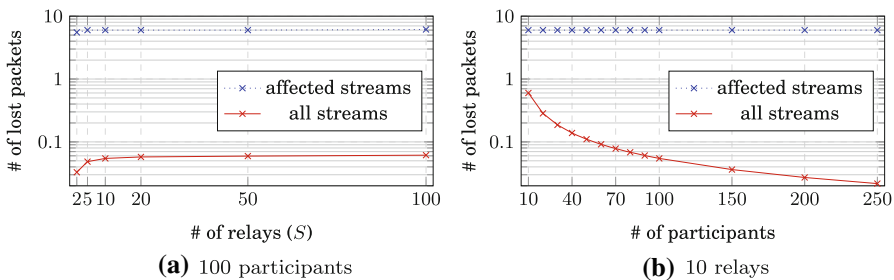


Fig. 11 Average number of audio packets lost during the redirection process

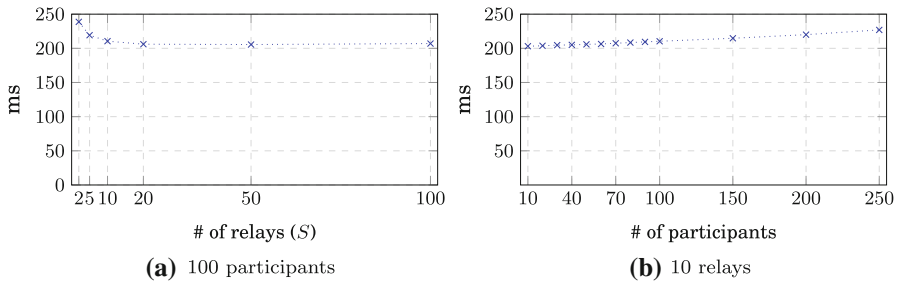


Fig. 12 Time required for the self-optimization technique

No packet loss occurs during the self-optimization technique, as the previous relay keeps forwarding data to a participant until he or she is re-directed to another relay. Occasionally, the participant may receive duplicate RTP packets. This is not an issue, since RTP packets contain a sequence number so duplicates can be discarded.

8 Conclusions

In this paper an autonomic platform for synchronous e-training in dispersed organizations is proposed. The platform uses standard protocols to deploy a mesh of RTP relays between geographically dispersed sites of the organization. Properties such as self-deployment, self-organization, self-healing and self-optimization are implemented to provide an efficient multimedia data delivery service between trainees located at their workplaces. Three virtual networks are established during the operation of the platform: a relay mesh to deliver data, a signaling network between participants and the RP, and a mesh control network between the relays and the RP to manage the organization of the mesh.

The use of standard protocols such as SIP for the session signaling and RTP for the data transport makes the platform inter-operable with other software and hardware conferencing solutions. Furthermore, the extensibility of the platform is guaranteed, and its modularity enables the portability of the platform to other data communication approaches without major changes.

The relay mesh used to transport multimedia data makes use of the available network resources efficiently. IP multicast is used where available, so participants communicate in real-time, using low network resources.

Moreover, the self-properties implemented in the platform make the management of the relay mesh automatic and transparent to the user, as well as robust in case of relay failures. The reorganization of the relay mesh is based on the joining and leaving of participants and relay failures.

The main drawback of the platform is its dependency on the RP. The RP is a centralized entity acting as a SIP focus, a SIP registrar and the manager of the relay mesh. A failure in the RP would prevent participants from joining synchronous e-training activities and changes in the organization of the relay mesh, but ongoing

activities can continue. This issue can be alleviated using fault-tolerant configurations of the RP.

Acknowledgments This research has been partially funded by the Spanish National Plan of Research, Development and Innovation under the project TIN2011-24903. Financial support (grant:UNOV-10-BECDOC-S) given by the University of Oviedo-Banco de Santander is also acknowledged.

References

1. Granda, J.C., García, D.F., Suárez, F.J., Peteira, I., Uría, C.: A multimedia tool for synchronous distance e-training of employees in geographically dispersed industries. In: Proceedings of the IASTED International Conference on Web-Based Education, pp. 1–8, Innsbruck, Austria (2008)
2. Diot, C., Dabbous, W., Crowcroft, J.: Multipoint communication: a survey of protocols, functions, and mechanisms. *IEEE J. Sel. Areas Commun.* **15**, 277–290 (1997)
3. Kompella, V., Pasquale, J., Polyzos, G.: Two distributed algorithms for multicasting multimedia information. In: Proceedings of the IEEE International Conference on Communications, pp. 343–349, Geneva, Italy (1993)
4. Zhang, X., Li, X.: Native multicast based videoconference service on CERNET. In: Proceedings of the International Conference on Communication Technology, pp. 1–4, Guilin, China (2006)
5. Diot, C., Neil, B., Bryan, L., Balensiefen, K.D.: Deployment issues for the IP multicast service and architecture. *IEEE Netw.* **14**, 78–88 (2000)
6. Yamamoto, L., Leduc, G.: Autonomous reflectors over active networks. In: Proceedings of the Symposium on Software Mobility and Adaptive Behaviour, pp. 125–146, York, UK (2001)
7. Nuño, P., Granda, J.C., Suárez, F.J., García, D.F.: Self-* in multimedia communication overlays. *Comput. Commun.* **36**, 817–833 (2013)
8. El-Sayed, A., Roca, V., Mathy, L.: A survey of proposals for an alternative group communication service. *IEEE Netw.* **17**, 46–51 (2003)
9. Yeo, C., Lee, B., Er, M.: A survey of application level multicast techniques. *Comput. Commun.* **27**, 1547–1568 (2004)
10. Čičić, T., Bryhni, H.: Multicast-unicast reflector. In: Proceedings of the Norsk Informatikk Konferanse, pp. 103–114, Kristiansand, Norway (1998)
11. Čičić, T., Bryhni, H., Sølve, S.: Unicast extensions to IP multicast. In: Proceedings of the International Conference on Protocols for Multimedia Systems, pp. 60–69, Krakow, Poland (2000)
12. Hladká, E., Holub, P., Denemark, J.: Teleconferencing support for small groups. In: Proceedings of the TERENCE Networking Conference, pp. 1–7, Limerick, Ireland (2002)
13. Hladká, E., Holub, P., Denemark, J.: User-empowered programmable network support for collaborative environment. In: Proceedings of the European Conference on Universal Multiservice Networks, pp. 367–376, Porto, Portugal (2004)
14. Patrikakis, C.Z., Despotopoulos, Y., Robotis, A.M., Lambiris, A.L.: PERIPHLEX: multicast delivery using core unicast distribution with peripheral multicast reflectors. In: Proceedings of the International World Wide Web Conference, p. 292, Budapest, Hungary (2003)
15. Lambrinos, L., Demetriou, E.: An adaptive live media streaming architecture. In: Proceedings of the IARIA International Conference on Advances in Multimedia, pp. 74–77, Athens, Greece (2010)
16. Maraviglia, G., Masi, M., Merlo, V., Licandro, F., Russo, A., Schembra, G.: Synchronous multipoint e-learning realized on an intelligent software-router platform over unicast networks: design and performance issues. In: Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation, pp. 1172–1179, Patras, Greece (2007)
17. Zhang, X., Li, C., Li, X.: Multi-party videoconferencing based on hybrid multicast with peer-forwarding. In: Proceedings of the IEEE International Conference on Parallel and Distributed Systems, pp. 812–816, Shanghai, China (2010)
18. Tan, S., Waters, G., Crawford, J.: A performance comparison of self-organising application layer multicast overlay construction techniques. *Comput. Commun.* **29**, 2322–2347 (2006)
19. Pendarakis, D., Shi, S., Verma, D., Waldvogel, M.: ALMI: an application level multicast infrastructure. In: Proceedings of the USENIX Symposium on Internet Technologies and Systems, pp. 49–60, San Francisco, USA (2001)

20. Liu, Z., He, G., Liu, Z.: Metrino: an application-level multicast system for real-time video communication. In: Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1265–1270, Wuham, China (2005)
21. Liu, Y., Guo, Y., Liang, C.: A survey on peer-to-peer video streaming systems. *Peer-to-Peer Netw. Appl.* **1**, 18–28 (2008)
22. Jannotti, J., Gifford, D., Johnson, K.L., Kaashoek, M.F., O’Toole Jr, J.W.: Overcast: reliable multicasting with an overlay network. In: Proceedings of the USENIX Symposium on Operating System Design & Implementation, 197–212, San Diego, USA (2000)
23. Chu, Y.H., Rao, S., Zhang, H.: A case for end system multicast. In: Proceedings of the ACM Sigmetrics, pp. 1–12, Santa Clara, USA (2000)
24. Banerjee, S., Bhattacharjee, B., Kommareddy, C.: Scalable application layer multicast. *Comput. Commun. Rev.* **32**, 205–217 (2002)
25. Castro, M., Druschel, P., Kermarrec, A.M., Nandi, A., Rowstron, A., Singh, A.: SplitStream: high-bandwidth multicast in cooperative environments. *ACM Oper. Syst. Rev.* **37**, 298–313 (2003)
26. Lua, E.K., Crowcroft, J., Pias, M., Sharma, R., Lim, S.: A survey and comparison of peer-to-peer overlay network schemes. *IEEE Commun. Surv. Tutor.* **7**, 72–93 (2005)
27. Qiu, T., Chen, G., Ye, M., Chan, E., Zhao, B.Y.: Towards location-aware topology in both unstructured and structured P2P systems. In: Proceedings of the International Conference on Parallel Processing, p. 30, Xian, China (2007)
28. Tran, D.A., Hua, K., Member, S., Do, T.: A peer-to-peer architecture for media streaming. *IEEE J. Sel. Areas Commun.* **22**, 121–133 (2004)
29. Yin, H., Lin, C., Zhang, Q., Chen, Z., Wu, D.: TrustStream: a secure and scalable architecture for large-scale internet media streaming. *IEEE Trans. Circuits Syst. Video Technol.* **18**, 1692–1702 (2008)
30. Zhang, X., Liu, J., Li, B., Peter-Yum, T.S.: CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming. In: Proceedings of the IEEE International Conference on Computer Communications, pp. 2102–2111, Miami, USA (2005)
31. Venkataraman, V., Yoshida, K., Francis, P.: Chunkyspread: heterogeneous unstructured tree-based peer-to-peer multicast. In: Proceedings of the IEEE International Conference on Network Protocols, pp. 2–11, Santa Barbara, USA (2006)
32. Pianese, F., Perino, D., Keller, J., Biersack, E.: PULSE: an adaptive, incentive-based, unstructured P2P live streaming system. *IEEE Trans. Multimed.* **9**, 1645–1660 (2007)
33. Liu, Z., Shen, Y., Ross, K.W., Panwar, S.S., Wang, Y.: LayerP2P: using layered video chunks in P2P live streaming. *IEEE Trans. Multimed.* **11**, 1340–1352 (2009)
34. Huang, N.F., Tzang, Y.J., Chang, H.Y., Ho, C.W.: Enhancing P2P overlay network architecture for live multimedia streaming. *Inf. Sci.* **180**, 3210–3231 (2010)
35. Wang, W., Chen, Y.: SmartPeerCast: a smart QoS driven P2P live streaming framework. *Multimed. Tools Appl.* **54**, 445–471 (2011)
36. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RTP: a transport protocol for real-time applications. IETF RFC 3550, <http://www.ietf.org/rfc/rfc3550.txt> (2003). Accessed 23 Sept 2013
37. Granda, J.C., Nuño, P., Suárez, F.J., Pérez, M.A.: E-pSyLon: a synchronous e-learning platform for staff training in large corporations. *Multimed. Tools Appl.* **66**, 431–463 (2012)
38. Granda, J.C., García, D.F., Nuño, P., Suárez, F.J.: An efficient networking technique for synchronous e-learning platforms in corporate environments. *Comput. Commun.* **33**, 1752–1766 (2010)
39. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: SIP: Session Initiation Protocol. IETF RFC 3261. <http://www.ietf.org/rfc/rfc3261.txt> (2002). Accessed 23 Sept 2013
40. Handley, M., Jacobson, V., Perkins, C.: SDP: session description protocol. IETF RFC 4566. <http://www.ietf.org/rfc/rfc4566.txt> (2006). Accessed 23 Sept 2013
41. Rosenberg, J.: A framework for conferencing with the Session Initiation Protocol (SIP). IETF RFC 4353. <http://www.ietf.org/rfc/rfc4353.txt> (2006). Accessed 23 Sept 2013
42. Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. Dependable Secur. Comput.* **1**, 11–33 (2004)
43. Elling, R., Pramanick, I., Mauro, J., Bryson, W., Tang, D.: Analytical reliability, availability, and serviceability benchmarks. In: Kanoun, K., Spainhow, L. (eds.) *Dependability benchmarking for computer systems*, pp. 23–33. Wiley, New York (2008)

44. Granda, J.C., Nuño, P., García, D.F., Suárez, F.J.: Security issues in a synchronous e-training platform. In: Proceedings of the IEEE International Conference on Availability, Reliability and Security, pp. 485–492, Vienna, Austria (2011)

Author Biographies

Juan C. Granda received his Ph.D. in Computer Science from the University of Oviedo, Spain, in 2008. He joined the Department of Computer Science and Engineering, University of Oviedo, where he is an Associate Professor. He has been involved in the research activities of the Computer Engineering Area, focusing on performance analysis of synchronous e-learning systems.

Pelayo Nuño received his M.S. degree in Computer Science from the University of Oviedo in 2009, where he is currently working on his Ph.D. degree. He has been involved in several research activities of the Computer Engineering Area. His research interests include multimedia networking and autonomic computing.

Daniel F. García is a full professor in the Department of Computer Science and Engineering at the University of Oviedo, Spain, where he leads the Computer Engineering Area. His current research interest is in the area of self-adaptive multimedia and real-time systems. During the last 20 years, he has led many research projects and coauthored 50 articles in journals and more than 130 papers in conferences and workshops. He is a member of the IEEE Computer Society.

Francisco J. Suárez is a professor in the Department of Computer Science and Engineering at the University of Oviedo, where he received his Ph.D. in 1998. His current research focuses on evaluation of networked multimedia systems and he has been working on several related projects as head of the Interactive Multimedia Services research team.