Using self-regulation to increase resilience in overlay networks for interactive multimedia communications

Pelayo Nuño[☆], Juan C. Granda, Francisco J. Suárez, Daniel F. García

Department of Computing Science, University of Oviedo
Campus de Gijón s/n, 33204, Spain

Abstract

Overlay networks are the underlying mechanisms which enable multipoint communications in interactive multimedia platforms. They must cope with constraints on heterogeneity, scalability and availability, making their management complex. Thus, several issues might affect the resilience of overlay networks, threatening ongoing communications. Overlay networks usually use self-organization and self-stabilization techniques to improve resilience. Techniques implementing other self-management properties such as self-regulation may also improve the resilience of the overlay. In this paper, the resilience of a reflector-based overlay network is improved with a self-regulation scheme based on audio transcoding and audio mixing. This scheme balances the workloads of the reflectors and saves network resources. As a result, the overlay is more resilient to failures in the reflectors. Extensive simulations have been carried out to compare various transcoding and mixing approaches, proving that self-regulation improves the resilience of the overlay network with little impact on end-to-end latency.

Keywords:
Overlay networks, resilience, self-regulation, self-stabilization

[☆]Corresponding author; Office 1.S.03, Edif. Departamental Oeste, Campus de Viesques, 33204, Gijón, Spain
Phone: +34 985 18 22 27, Fax: +34 985 18 19 86
Email addresses: {nunopelayo,jcgranda,fjsuarez,dfgarcia}@uniovi.es

1. Introduction

Nowadays, interactive multimedia is present in many activities such as conferencing, education, training and leisure, imposing severe constraints on the underlying platform used to support multimedia communications. These are mainly related to heterogeneity, scalability and availability of communications. The core of a multimedia communication platform is an overlay network, which is composed of many members [1]. The overlay network can provide services previously unavailable in the existing network. A Resilient Overlay Network (RON) includes mechanisms to face unforeseen events, automatically recovering from failures in different members of the overlay [2] within a short period of time [3], thus providing a reliable data delivery service.

Many RONs rely on the Autonomic Computing (AC) paradigm. The aim of an AC system is self-management, *i.e.*, the ability of the system to manage itself while hiding its complexity to users [4]. Self-management is an inherent part of overlay networks to cope with several key aspects such as performance, reliability, deployment, robustness and maintainability. The members composing an overlay network can be distributed, performing independently, without awareness of the whole system; they must manage themselves, showing neither changes in their behavior nor degradation in the quality of service provided to users. The AC paradigm can also improve the resilience of overlay networks. For example, an autonomic mechanism can be used to detect network disruptions, recovering from them and dynamically discovering optimal network paths among members [5].

The AC paradigm encompasses several autonomic properties. Self-organization, which is the capacity of a system to interact with the environment without being guided, is the most widely used for improving resilience in RONs [6]. This property is usually implemented with distribution data schemes using different paths [7], so the forwarding effort is balanced and fairness among network links is guaranteed. According to such a scenario, a failure in a network link only compromises the path to which it belongs and data will be delivered through the remaining non-faulty paths. The exclusion of failing members leads to another autonomic property of RONs: self-stabilization.

There are other self-management properties, less common in overlay networks, which RONs may take advantage of to improve their resilience. Self-regulation is a notable example. Self-regulation refers to the ability of a system to adjust or reconfigure its parameters to guarantee the quality of service depending on the operational conditions and the environment [8]. Self-regulation maintains fairness among concurrent applications and clients, and guarantees a level of performance [9].

Media transcoding and media mixing techniques are well-known examples of self-regulation in the context of multimedia communications. Transcoding implies the re-coding of a multimedia stream from one format into another, so a data stream originally encoded at a high bit rate can be adapted to cope with channel or client resource limitations [10]. In addition, several streams are combined to generate a new stream when media mixing is performed [11], so the number of multimedia streams in the network is decreased. Both techniques have proved to improve the performance of a multimedia communication system [12, 13].

In this paper, audio transcoding and audio mixing approaches are proposed as a means to increase the resilience of an overlay network. The overlay includes various self-management properties such as self-organization, self-optimization, self-stabilization and self-regulation. Audio streams can be mixed and transcoded, so the overlay adapts dynamically to the conditions of the members and the underlying networks. This increases self-regulation and self-stabilization when a failure occurs in any member of the overlay. The capacity for self-stabilization of the overlay network is analyzed under different transcoding and mixing scenarios.

The remainder of the paper is organized as follows. In Section 2, related work on self-managed RONs and audio transcoding and mixing is discussed. The architectural design of the RON is presented in Section 3. The self-regulation scheme is described in Section 4. This is evaluated in Section 5 and the results exposed and discussed in Section 6. Finally, Section 7 contains the concluding remarks.

2. Related Work

The AC paradigm is introduced in [4] and formulates the fact that autonomic systems have four basic properties, usually referred to as self-properties [14]: self-configuration, self-optimization, self-healing and self-protection. These self-properties have been further divided into more fundamental autonomic properties [15] such as self-stabilization [16], self-organization [17], self-immunity and self-containment [18]. Many of them can be successfully applied to multimedia communication systems [19].

The main purpose of an autonomic system is self-management. Some of the self-properties related to the AC paradigm may help build resilient systems. Specifically, RONs may rely on self-organization and self-stabilization to cope with failures. A survey of overlay networks providing resilience based on self-organization techniques is presented in [6], classifying them in three approaches: cross-link, in-tree and multiple-tree redundancy techniques. These approaches use path diversity to ensure that data reaches all the members of the overlay. This leads to non-topological, dependent communication. A resilient member is achieved with the help of several adjacent members and backup paths between them in cross-link and in-tree redundant techniques. Cross-link redundancy connects random members [20], while in-tree redundancy organizes members in cluster-based trees and builds alternative paths, linking each cluster with members of other clusters [21]. In contrast, several overlapped data distribution trees are built in multiple-tree redundancy [22].

Self-regulation can also be useful to improve the resilience of RONs. Self-regulation is related to self-optimization. In the context of overlay networks, self-regulation can be identified in systems with members capable of modifying both the media encoding and the composition of the content delivered on the fly. Thus, the presence of bottlenecks and overloads in the overlay is minimized. Various approaches can be used to implement self-regulation in multimedia communication systems: scalable coding, mainly used for video streams, media transcoding and media mixing. This paper focuses on the last two approaches, applied to audio communications.

2.1. Audio transcoding

Audio transcoding is a technique to encode audio streams from one format into another. Audio transcoding has typically been used for interoperability purposes, making it possible to build communication platforms when the endpoints use different audio encodings [23, 24]. However, audio transcoding can also be used to save network resources by converting high-bitrate streams into low-bitrate streams [25, 26]. A disadvantage of audio transcoding is a slight increase in latency [27], which could endanger its suitability for interactive multimedia communications.

Audio transcoding can be performed according to several architectures. An active node is deployed between the media server and the endpoints in a cluster-based architecture [28]. This node masks several computing nodes at the back-end to build a transcoding cluster. This architecture is mainly used for media streaming, so that unlike the strict constraints of interactive communications, latency is not critical. A distributed architecture to perform audio transcoding in peer-to-peer overlay networks is introduced in [29]. Overlay nodes transcode audio of joining nodes according to requirements of the latter. Finally, a proxy-based architecture is also possible, transcoding between two network domains, for example wired and wireless [25]. In this case, a transcoder proxy is placed in each domain. A similar approach can be implemented in reflector-based overlays to transcode both audio and video [30].

2.2. Audio mixing

Audio mixing enhances the performance of multipoint conferencing systems combining various media streams into a new stream [13]. The advantages of audio mixing can also be applied to interactive multimedia communications [31]. Several architectures for audio mixing in conferencing systems are discussed in [32]: centralized, endpoint, hierarchical, distributed partial mixing (DPM) and distributed mixing (DM). Some of these are not suitable for interactive multimedia communications. Centralized mixing leads to bottlenecks in the network for highly distributed scenarios. Endpoint mixing increases the workload at each endpoint, so reliability and robustness are threatened. In any case, latency of communications and interarrival jitter are severely penalized in large distribution trees.

However, two of the aforementioned audio mixing architectures can be successfully applied to interactive multimedia communications: DPM and DM. Both alternatives deploy a set of distributed mixers across the overlay in order to reduce the number of concurrent streams. In DPM, mixers select a random subset of streams to mix from all those received. This approach can be implemented in audio streaming [33] or video customization according to user scoring [34]. DPM may introduce significant latency when the number of hops increases. Nevertheless, latency is lower when the overlay network is deployed in a full-mesh topology, as data between endpoints need traverse fewer hops.

DM imposes an architecture divided in two levels [35]. The first level is composed of mixers and the second level is composed of endpoints without mixing capabilities. Each mixer processes all the streams generated from a subset of endpoints. The forwarding effort of mixers can be alleviated by adding distribution nodes between endpoints and mixers [36]. This approach is improved in [37], where a cross-link RON is built including: self-organization of mixers (using merging, splitting and migrating actions) and self-containment (by applying role redundancy to the mixers with back-up members). The DM architecture has been recently implemented in voice communications for massive multiplayer games [38], conferencing [39], and mobile ad-hoc networks [40].

3. Overlay Network Architecture

An overlay network for interactive multimedia communications is described in [41]. This overlay provides an efficient multimedia delivery service by automatically organizing itself based on the joining and leaving of participants to an interactive multimedia activity (video conference, e-meeting, synchronous e-learning), hereinafter referred to as activity. The overlay is composed of various entities as shown in Fig. 1: one Rendezvous Point (RP), one Binary Floor Control Protocol (BFCP) server, several or no Real-time Transport Protocol (RTP) relay servers and the participants. These entities are organized in four virtual planes: the signaling plane, the relay mesh, the mesh control plane and the floor control plane. The participants interact with all these entities of the overlay during the activity.

The RP acts as a Session Initiation Protocol (SIP) focus of a tightly coupled conferencing service and a SIP registrar server in the signaling plane. It provides a standard mechanism for the authorized participant to gain access to the activity using SIP. This protocol is used to establish and tear down the multimedia RTP sessions associated with an activity. These sessions are described using the Session Description Protocol (SDP). Each participant must establish a standard SIP dialog with the RP to join the activity. As a result, the participant is assigned a relay, so the multimedia data streams are interchanged between the participant and the relay.

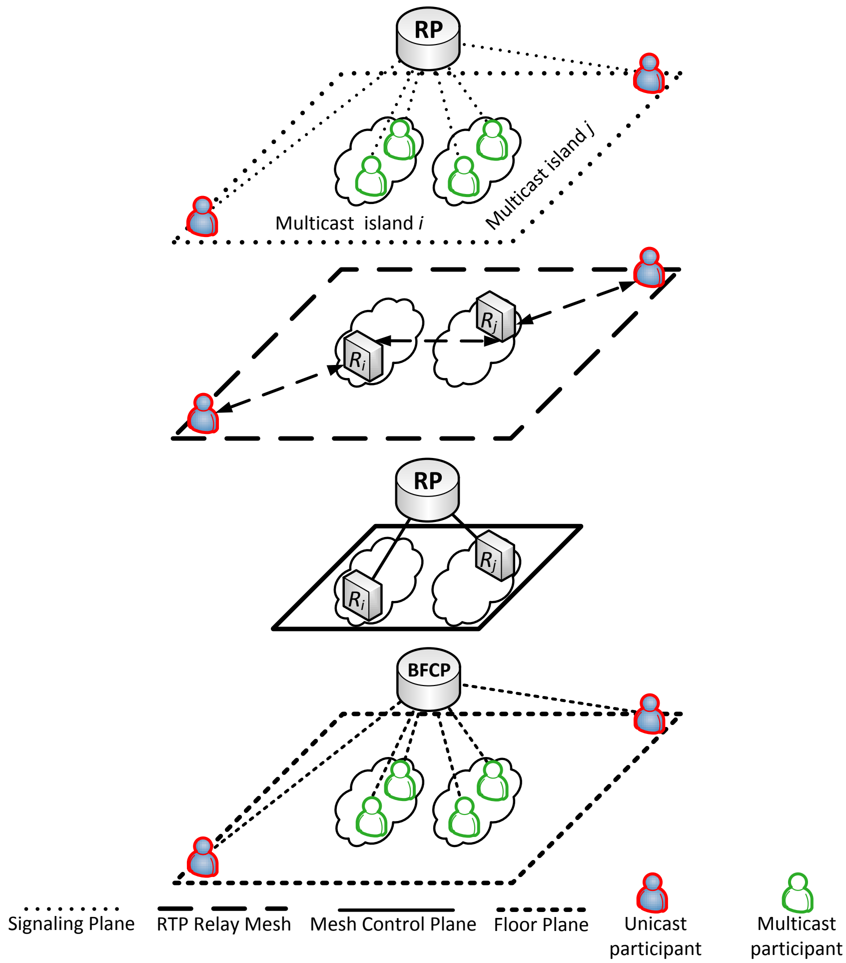


Figure 1: Architecture of the overlay network

All the multimedia data is conveyed using RTP through the relay mesh. It is assumed that IP multicast is partially available, so the relay mesh interconnects several multicast islands where participants are scattered. A relay is located in every multicast island acting as a reflector forwarding traffic between the participants of the multicast island and the rest of participants in the activity. The relays are connected in a full-mesh topology to minimize the network latency of communications. This means that multimedia streams traverse a maximum of 2 relays. Data is interchanged between participants and relays using IP multicast, while a relay forwards data to the rest of the relays using unicast connections. This design has proved to be highly efficient [42].

Occasionally, participants are located outside a multicast island and they must be assigned a relay to join the activity. In this case, multimedia streams are interchanged between the participant and the relay using unicast communications.

The mesh control plane is established between the relays and the RP using TCP connections. The relays composing the mesh may vary throughout the activity. The RP is responsible for the re-organization of the relay mesh as participants join and leave. A relay must contact the RP as soon as it starts, so the RP is aware of the different multicast islands and can associate relays with participants in their multicast islands. Thus, the RP can change the relay mesh organization including a new relay when the first participant of its multicast island joins, or excluding an active relay when the last participant of its multicast island leaves. These changes are reported to all the relays in the activity, so they know their peer relays in the relay mesh.

Finally, the floor control plane involves the participants and the BFCP server. This server is responsible for developing the floor control policy of an activity, granting and revoking floors as instructed by the moderator, so the number of concurrent media streams can be limited. The moderator can be a participant within the activity or an external user. All the participants establish a TCP connection with the BFCP server for the exchange of floor control messages.

3.1. Self-management properties

The overlay network includes a self-organization technique that ensures a minimum number of streams interchanged among multicast islands. The RP maintains a list of registered relays, associating them with the identifier of the multicast island in which they are located. Since the relay mesh only involves active relays, the two basic processes involved in the optimization of the relay mesh are the inclusion and exclusion of relays.

A relay is included in the mesh when it becomes active. This occurs whenever the first participant of its multicast island joins the activity. The RP activates the relay so it begins forwarding data from the activity to the new participant and the data generated by the new participant to the activity. The rest of the relays in the mesh must be informed when a relay becomes active so they can forward data to it.

On the other hand, a relay is excluded from the relay mesh when it becomes inactive. This occurs whenever the last participant remaining in

```
procedure RecoverFromRelayFailure(activity, relay)
    relays ← GetActiveRelays(activity)
    for all participant ∈ GetServedParticipants(activity, relay) do
        newRelay ← ChooseAlternativeRelay(relays, relay, participant)
        DeAssociate(activity, participant, relay)
        ReDirect(activity, participant, newRelay)
    end for
end procedure

procedure ReDirect(activity, participant, newRelay)
    sdp ← CreateSDP(activity, participant, newRelay)
    if ReInvite(participant, sdp) then
        Associate(activity, participant, newRelay)
    else
        Disconnect(participant)
        RemoveFromActivity(activity, participant)
    end if
end procedure
```

Figure 2: Self-stabilization algorithm

its multicast island leaves the activity. The RP sends an exclusion message to the relay, so the relay disconnects from the relay mesh. In addition, a removal message is sent by the RP to all the active relays to notify that the relay has become inactive, so they stop forwarding data to it.

Furthermore, the overlay network includes a cross-link redundancy self-stabilization technique to increase the overall resilience. Whenever a relay goes down, the RP redirects the participants of the relay to other active relays in the mesh. In this way, the participants still send and receive traffic to and from the ongoing activity. The algorithm used to redirect participants to alternative relays is shown in Fig. 2. Several heuristics may be used to choose the alternative relay based on the proximity with the original relay, minimum latency or current load, to name a few.

The self-stabilization technique allows for the continuity of the service in spite of relay failures. In order to balance the workload among the relays in the mesh, the RP uses an estimation of the available bandwidth in each relay to redirect participants. This technique collaborates with the self-organization technique previously described to provide a resilient and efficient data delivery service.

When the relay is up again and registered with the RP, the reassignment process is undone. The RP maintains a list of the changes made in the relay mesh, so it is able to undo any of them. In this case, the self-organization technique is applied to redirect participants to their original relay.

However, the redirection process has some drawbacks. Redirected participants must use unicast to communicate with the relays, so the consumption of network resources increases. In addition, the overlay network does not have proactive mechanisms to predict and avoid overloads or bottlenecks that may jeopardize the overall performance.

4. Proposed self-regulation scheme

Whenever a participant is redirected, the relay must forward all the media streams in the activity to these participants using unicast, so the impact of redirected participants is significant. This can be mitigated by using self-regulation techniques based on audio transcoding and audio mixing. An incomplete self-regulation scheme based on audio transcoding has already been proposed [43]. The scheme proposed in this paper extends the original one, including audio mixing. This scheme is applied to the overlay described in the previous section. The aim is to reduce the consumption of network resources in the relays when data is forwarded, so the self-stabilization ability of the overlay network increases.

4.1. Audio transcoding approaches

Audio transcoding can be used to reduce the bitrate of the streams forwarded to or coming from redirected participants. This bitrate reduction should be a tradeoff between the saved network bandwidth and the resulting audio quality. Next, several approaches to perform self-regulation based on audio transcoding are described. To have a better understanding, let α and β be two media codecs where the bitrate used for an α stream is greater than that used for a β stream (this relation may also stand for audio quality). A multicast participant refers to a participant located in the multicast island of its relay, while a unicast participant is located outside the multicast island of its relay or has been redirected to another relay.

a) *No transcoding.* No transcoding is applied to the audio streams. As shown in Fig. 3, audio streams are sent and received by redirected participants with no modification. This scheme is mandatory when participants support only one codec.

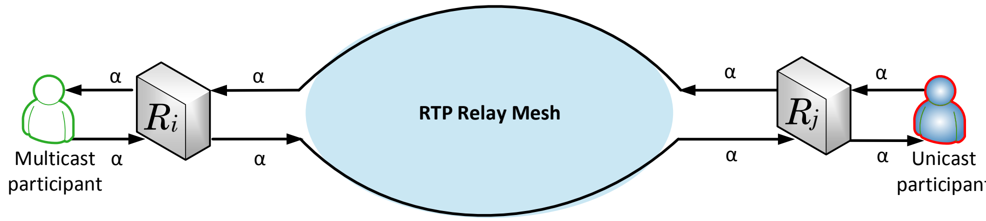


Figure 3: No transcoding

b) *Downstream transcoding.* Relays transcode all the streams forwarded to unicast participants to save bandwidth as illustrated in Fig. 4. However, relays do not transcode the streams coming from unicast participants (codec α). This is useful when some of the rest of the participants support only one codec or when a decrease in the quality of the streams sent by unicast participants is inadmissible.

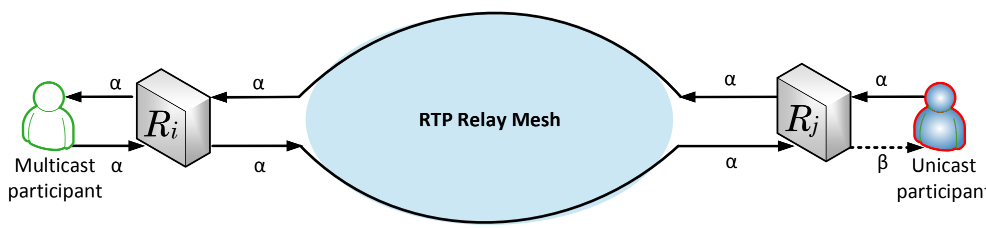


Figure 4: Downstream transcoding

c) *Full transcoding.* This scenario extends the previous one as shown in Fig. 5. Unicast participants use only the low bitrate codec, but relays transcode all the streams forwarded to and coming from unicast participants, so the rest of the participants need only support codec α. This approach saves more bandwidth by decreasing the quality of the audio streams sent and received by unicast participants. Although the transcoding of the streams sent by unicast participants from codec β to codec α does not increase the quality of the streams, the use of a single codec within the relay mesh makes it possible to carry out mixing and accounting operations over the streams uniformly.

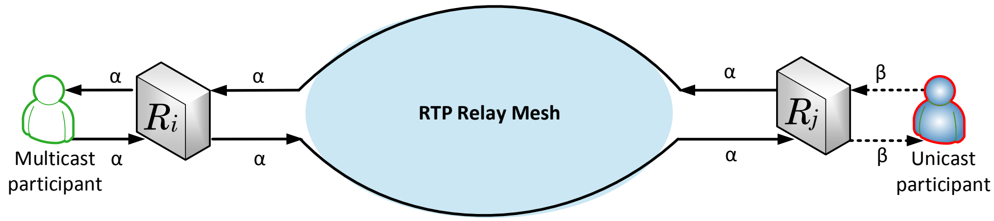


Figure 5: Full transcoding

d) *Enhanced downstream transcoding*. This approach, presented in Fig. 6, is similar to downstream audio transcoding, but unicast participants use codec β to encode their audio media streams. Thus, unicast participants only process β streams. This alternative is the most efficient in terms of the overall network bandwidth consumed. However, different encoded audio streams flow throughout the relay mesh, making the mixing or accounting of the streams cumbersome. In addition, all the participants must support codec α and β.

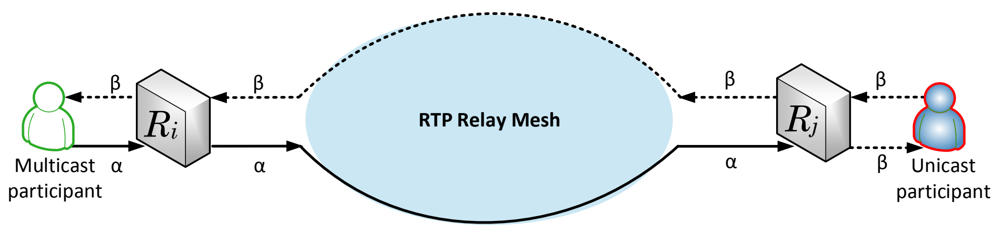


Figure 6: Enhanced downstream transcoding

e) *Deferred full transcoding*. This alternative is a modification of full transcoding where the streams coming from the unicast participants, encoded with codec β, are transcoded to codec α just before forwarding them to the receivers. This alternative saves bandwidth in the relay mesh, as relays forward the streams from the unicast participants in a low bitrate format. This process is illustrated in Fig. 7. In this case, unicast participants use only codec β, while the rest of the participants use only codec α.

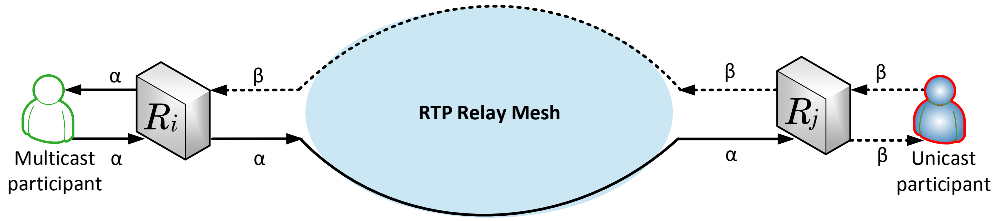


Figure 7: Deferred full transcoding

4.2. Audio mixing approaches

Audio mixing can be used to decrease the number of streams in the relay mesh. The mixing approach should consider a tradeoff between the saved network bandwidth and the number of streams mixed, because the mixing process may lose some information from the individual streams to mix (merge). The recommended upper bound number of concurrent audio streams is two, to prevent the listeners from having to make a notable effort [44], assuming that all the audio streams contain useful information (talkspurts). However, interactive communications are usually self-moderated in the sense that speakers tend not to overlap with other speakers, so mixing more than two concurrent streams (talkspurts in a few streams and silence in the rest of the streams) is reasonable under these conditions.

Next, several approaches to perform self-regulation based on distributed audio mixing are discussed. To have a better understanding, let α_n be an audio stream generated by participant n, and let $\alpha_{m,n}$ be a stream mixed by the relay that merges the audio streams generated by participants m and n into one stream.

a) *No mixing.* No mixing is applied to the audio streams. As shown in Fig. 8, the number of audio streams received by each participant who is not sending data matches the number of concurrent audio sources.

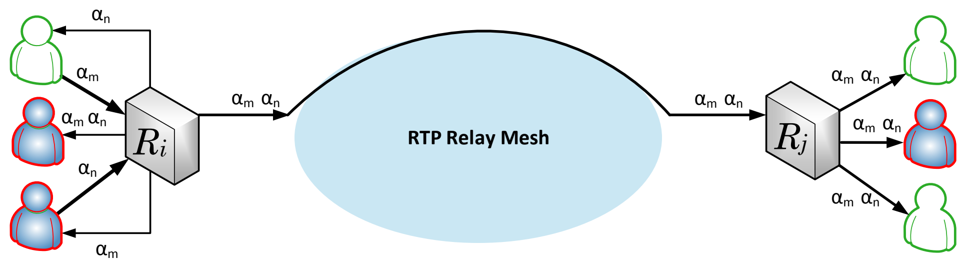


Figure 8: No mixing

b) *Downstream mixing.* The relays mix all the streams forwarded to unicast participants into one stream in order to save bandwidth, as shown in Fig. 9. The participants of the multicast islands of the relays still receive the audio streams separately. This reduces the network resources required in the links of the relays to forward multimedia data to unicast participants, as only one stream is sent to each unicast participant.

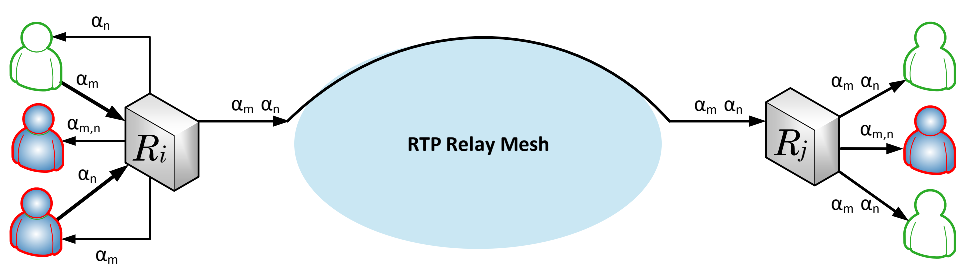


Figure 9: Downstream mixing

c) *Full mixing.* In this scenario all the audio streams are mixed once they are received at a relay. Thus, only mixed streams flow through the relay mesh and the individual original streams are limited to the multicast islands where they are generated, as shown in Fig. 10. This approach saves more network bandwidth than the previous one. However, it also imposes extra workload on the relays.

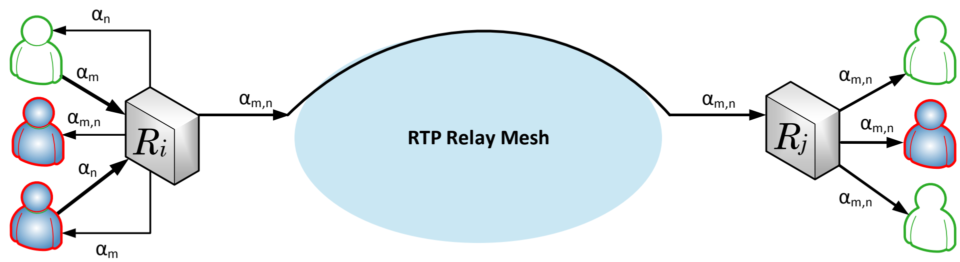


Figure 10: Full mixing

Finally, audio transcoding and mixing approaches could be combined into a hybrid approach by transcoding every stream generated after mixing the original streams to codec β. However, hybrid approaches are not considered in this work.

5. Experimentation

The performance of the self-regulation techniques is analyzed using three types of tests. Firstly, the use of the aforementioned audio transcoding and mixing approaches has some impact at the relays of the RTP mesh by decreasing the amount of network resources used. For this reason, some of the tests carried out focus on the network resources used by relays. Secondly, the bandwidth saving may increase the self-stabilization ability of the overlay network and consequently its resilience. Finally, end-to-end latency is critical in interactive multimedia communication services, so the effect of transcoding and mixing techniques on end-to-end latency is assessed.

Three transcoding approaches were used during the tests. The approach where no transcoding is performed is used as a reference for comparison with the full transcoding and enhanced downstream transcoding approaches. The other two transcoding approaches are not considered, since they are slight modifications of the former. In addition, all the mixing approaches are considered and compared to the reference.

First, to analyze the bandwidth consumed by the relays the whole network overlay is modeled using the ns-3 simulator. Data to build the model was collected from real activities developed with an e-training platform within a corporate scope [41]. The model simulates an overlay network deployed in several multicast islands where IP multicast is available, with an RTP relay

Table 1: Simulation settings

Waiting time to join (s)	λ_w	1/45
Audio activation interval (min)	(μ_a,σ_a)	(25,3.3)
Audio stream duration (s)	(μ_{al},σ_{al})	(15,2.6)
Activity duration (min)	d	30
Bandwidth available at relays (Mbps)	B_i	20
Link site delay to the WAN (ms)		25
RP link bandwidth (Mbps)		10
Number of concurrent audio streams (max)	f	4
Primary codec (kbps)	α	15.2
Secondary codec (kbps)	β	7.4
Maximum mixing ratio (i/o streams)	r	4:1

placed in every multicast island, and an RP managing the overlay. Table 1 summarizes the parameters considered during these tests. A Wide Area Network (WAN) connects every island through 20 Mbps network links with a propagation delay of 25 ms. The data rate of the network link connecting the RP to the WAN is 10 Mbps and introduces no delay in communications.

A participant is simulated as an entity that generates an audio stream. A participant waits before joining the activity. The waiting time is simulated using an exponential random variable $Exp(\lambda_w)$. Once in the activity, the participant remains joined until the end. Real activities are usually moderated in order to avoid excessive consumption of network resources due to many overlapped multimedia streams. Therefore, a maximum of f audio streams are allowed in an activity simultaneously. The participants are granted the use of the audio channel in FIFO order.

The audio stream of a participant is activated regularly. The interval between activations and the duration of the audio stream are assumed to be normal random variables. Thus, a participant generates an audio stream of duration $\mathcal{N}(\mu_l,\sigma_l^2)$ after an elapsed time $\mathcal{N}(\mu_a,\sigma_a^2)$. An audio stream is a simulated CBR iLBC audio stream with a packetization time of 20 ms and a bit rate of 15.2 kbps (codec α), which results in a network bandwidth consumption of 31.2 kbps after adding the overhead generated by the protocol headers (RTP/UDP/IP). When transcoding takes place, iLBC audio streams are transcoded to AMR 7.4 kbps (codec β), leading to a network bandwidth consumption of 24 kbps. Both codecs achieve toll quality when encoding voice. In relation to the mixing approaches, a relay uses an r mixing ratio which implies a maximum of 4 inputs streams per output stream.

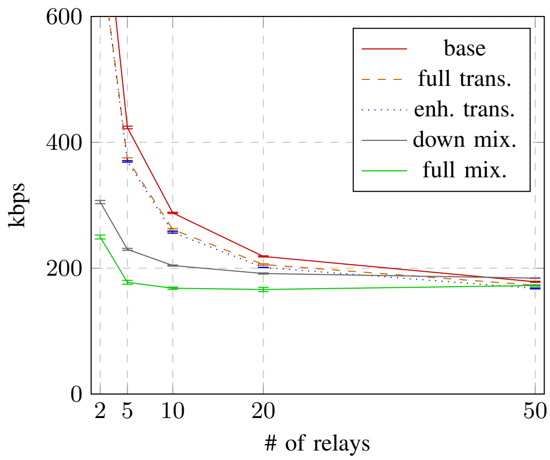
Second, several tests are carried out to evaluate the impact of the self-regulation scheme on the self-stabilization ability of the overlay. The self-stabilization technique used to redirect participants when a relay fails cannot cope with indefinite relay failures if the relays do not recover. Redirected participants communicate using unicast, so the resources consumed significantly increase after a relay failure. The number of participants that can be redirected varies depending on the transcoding and mixing approaches used, so various simulations are performed to analyze this relation. The self-stabilization technique of the overlay redirects each participant to the relay which is serving the lowest number of participants.

Subsequent random relay failures in an established overlay network during an activity are simulated until a relay reaches its maximum available bandwidth due to redirected participants. In this way, the threshold of redirected participants and recovered sites before the overlay exhausts its resources can be found. In these tests, a worst case scenario is assumed, since all the audio streams are continuously granted (f streams), so the effect of each approach can be clearly identified. Furthermore, for these tests the data rate of the links connecting each multicast island to the WAN is simulated using a normal random variable $B(10{,}3)$ Mbps with a lower bound threshold of 1 Mbps. The rest of the parameters are shown in Table 1.

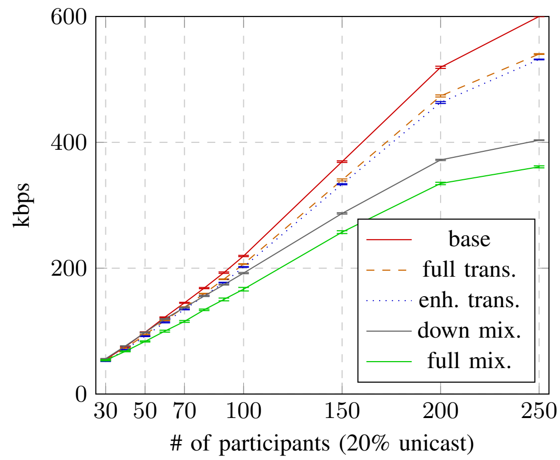
Finally, an Astersik PBX is used as a relay to analyze the delay introduced in the end-to-end latency by the transcoding and mixing techniques. The SIPp traffic generator is used to simulate participants, so a voice over IP call is established with the Asterisk PBX for each participant. The PBX is reponsible for transcoding the iLBC audio streams to AMR audio streams when using transcoding and mixing various audio streams into one stream when using mixing. The Asterisk PBX is run on a single-core virtual machine with an Intel Core i7-7700, 1 GiB of RAM and SSD storage. In these tests, all the audio streams are continuously granted and there is no limit on the mixing ratio.

6. Results

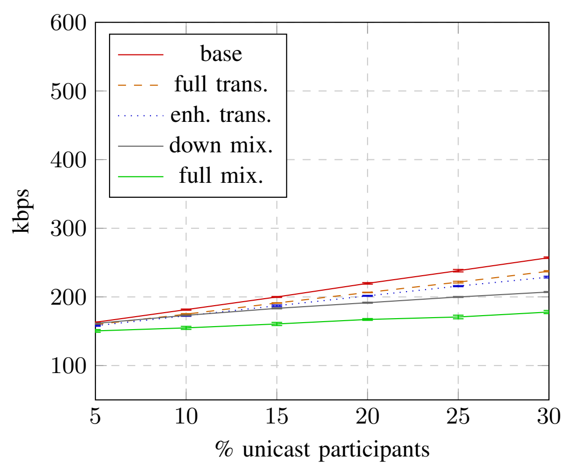
The amount of network resources used by relays is depicted in Figure 11. Confidence intervals with $\alpha = 0.05$ are also shown. Three parameters are varied during the tests: the number of relays in the mesh, the number of participants in the activity and the percentage of participants using unicast to communicate with their relays.



(a) Activity with 100 participants (20% unicast)



(b) Activity with 20 relays



(c) Activity with 100 participants and 20 relays

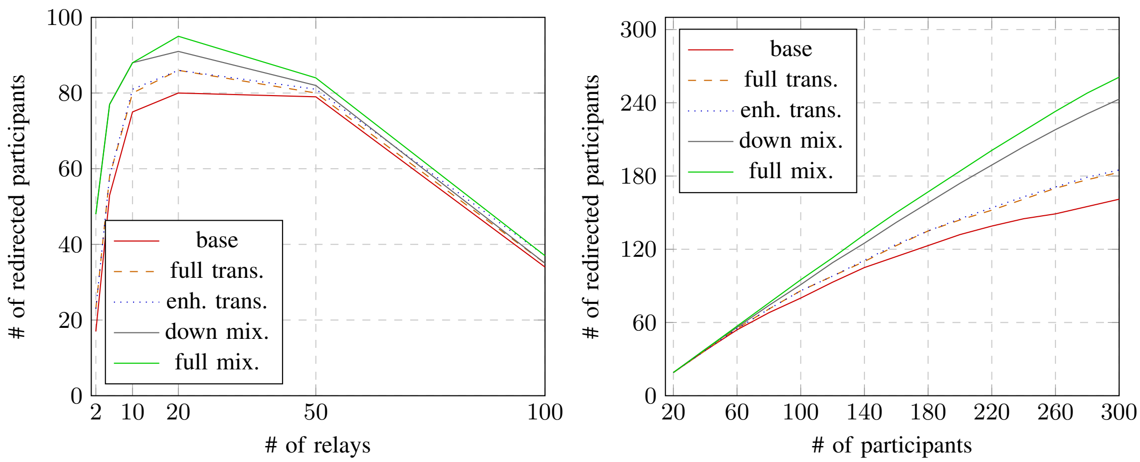
Figure 11: Average network bandwidth consumption in the network links of the relays

As can be observed in Fig. 11a, the performance of audio transcoding approaches is higher than the performance of the base scenario (no transcoding or mixing) for a low number of relays in the mesh. The performance of the mixing approaches is even better in the same conditions. However, all the scenarios tend to converge as the number of relays increases. The reason is twofold. First, the larger the number of relays, the less likelihood of a relay hosting more than one data source in its multicast island at the same time, so the performance of mixing approaches decreases with the number of relays. Second, RTCP packet replication increases, so RTCP becomes the main traffic in the mesh for high numbers of relays in the mesh. Mixing approaches further increase RTCP traffic, since each relay becomes a mixing source generating its own RTCP packets. This explains why transcoding approaches save more bandwidth than mixing approaches for a high number of relays. The higher the number of participants, the stronger this tendency.

Figure 11b examines the influence of varying the number of participants in an activity with 20 relays. A linear tendency can be identified in all the scenarios until the number of participants reaches a threshold where the increase in the bandwidth consumption becomes asymptotic due to the limitation of concurrent audio streams. For a high number of participants, the number of unicast participants increases, so the mixing approaches are able to save more bandwidth compared to the rest of scenarios, since a lower number of streams flows through the relay mesh. The bandwidth consumption grows at the same pace in the base and transconding scenarios, although the bandwidth used in the latter is lower due to the saving in RTP traffic. For a low number of participants the mixing approaches consume slightly more resources than the rest because of the extra RTCP traffic introduced by the relays. As the number of participants increases (and so the number of unicast participants) the mixing approaches become more efficient. This tendency is clearly seen in Fig. 11c where the influence of unicast participants is examined. Again, the mixing approaches have a better performance than the rest for a high number of unicast participants since the pace of the growth of the bandwidth consumption is lower than for the rest of scenarios.

Figure 12 shows the maximum number of participants that can be redirected when subsequent relay failures occur in an activity, with all the relays serving the same number of participants and without initial unicast participants. Figure 12a shows the maximum number of participants that can be redirected when increasing the number of relays in the mesh for an activity with 100 participants, while Fig. 12b shows the same when the number of participants increases in an activity with 20 relays.

As is illustrated in Figure 12a, the ability to redirect participants when relays fail increases with the number of relays in the mesh, since the number



(a) Activity with 100 participants

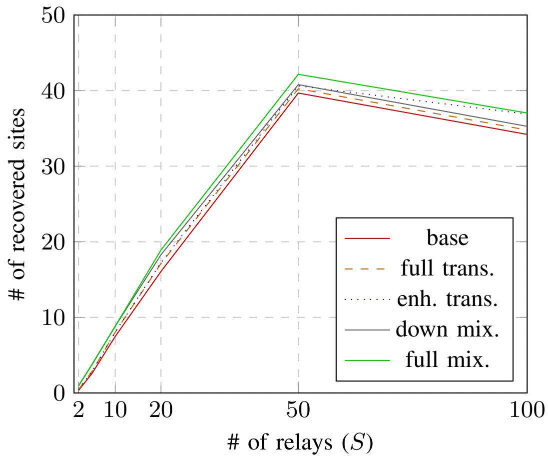
(b) Activity with 20 relays

Figure 12: Maximum number of redirected participants after subsequent relay failures

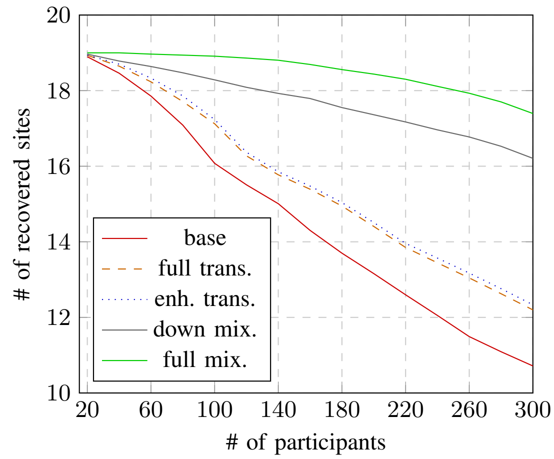
of participants per relay is lower. However, the number of redirected participants decreases for more than 20 relays as the bandwidth consumption increases due to packet replication. Therefore, audio mixing approaches allow for redirecting a higher number of participants for a small number of sites. In contrast, for an activity with 100 relays there is no difference between full mixing and enhanced transcoding, and with a larger number of relays the improvement is higher in the latter approach. Again, this is due to the lower likelihood of a relay hosting concurrent data sources in its multicast island at the same time when the number of sites increases. For this reason, in a scenario where there are no senders or only one sender in the same multicast island, enhanced transcoding is the most efficient approach to increase the self-stabilization ability of the overlay.

On the other hand, when increasing the number of participants the ability to redirect them also increases, as can be seen in Figure 12b. However, the growth is higher when self-regulation techniques are applied, and especially when using audio mixing approaches. Full mixing performs better than downstream mixing, while differences between the two audio transcoding approaches are slight. Therefore, Figure 12 shows that self-regulation techniques improve the self-stabilization ability of the overlay, particularly for activities with a dispersion up to 20 relays.

Figure 13 shows the maximum number of sites (relays) that can be recov-



(a) Activity with 100 participants



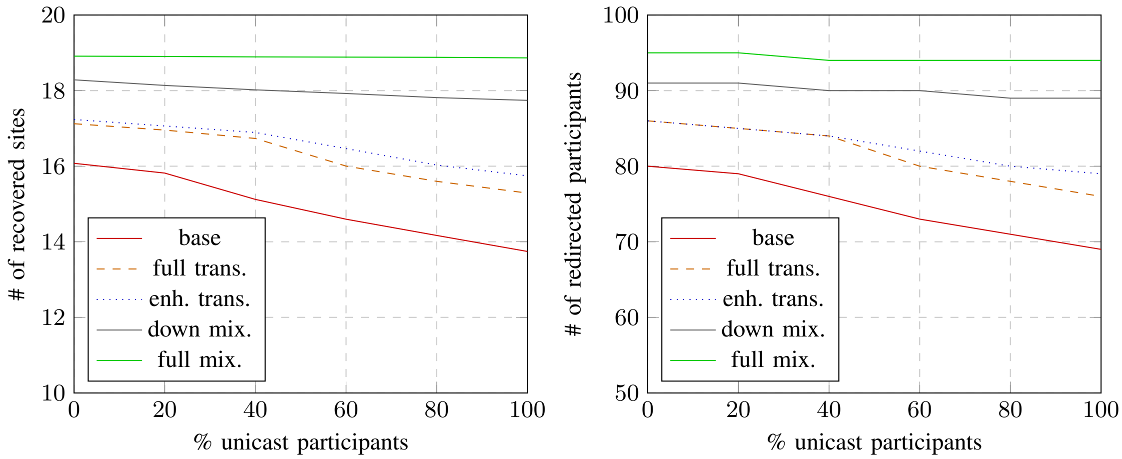
(b) Activity with 20 relays

Figure 13: Maximum number of sites recovered after subsequent relay failures

ered by redirecting their participants when subsequent relay failures occur in a scenario similar to the previous test. Fig. 13a shows this as a function of the number of sites for an activity with 100 participants. According to the results, self-stabilization is slightly enhanced when self-regulation approaches are used. This is especially observable in cases where the number of relays is high. Full-mixing is again the better approach, but the performance of the enhanced transcoding approach is similar for the maximum number of relays.

Fig. 13b shows the number of recovered sites as a function of the number of participants for an activity with 20 relays, which is the site dispersion threshold previously determined in Figure 12. In this case, the effect of using self-regulation techniques is noticeable, especially for audio mixing approaches. As the number of active relays decreases due to failures, the likelihood of a relay hosting several data sources is higher, so mixing approaches are able to save more bandwidth in the RTP relay mesh, postponing overlay saturation.

The influence of unicast participants when applying different self-regulation techniques is analyzed in Fig. 14. The simulated activity involves 100 participants, organized in 20 sites with all the relays serving the same number of participants at the beginning of the activity. Figure 14a depicts the maximum number of relays that can be recovered when increasing the number



(a) (b)

Figure 14: Maximum number of sites recovered and redirected participants after subsequent relay failures according to initial unicast participants

of initial unicast participants, while Fig. 14b shows the maximum number of participants that can be redirected in the same scenario.

As can be seen, self-stabilization is improved when using self-regulation techniques. Audio mixing approaches obtain the best performance again. In fact, the full mixing approach shows no degradation in the self-stabilization ability of the overlay according to the increase of the initial number of unicast participants. Regarding audio transcoding, the enhanced transcoding approach performs slightly better than full transcoding due to a saving in bandwidth consumption in the links of the relays.

Figure 15 depicts the delay introduced by the transcoding of iLBC audio streams into AMR streams when increasing the number of participants served by a relay. As can be seen, the relay can easily transcode up to 80 audio streams simultaneously with low delay despite its limited resources (one CPU core and 1 GiB of memory). Once the relay reaches saturation, the delay introduced by the transcoding process skyrockets, which makes the end-to-end latency unacceptable for real-time communications.

In the case of mixing, the relay uses a worker thread to mix all the audio samples of the incoming streams into a single audio stream. Since iLBC streams use an audio frame of 20 ms, the worker thread is triggered every 20 ms. This is the ideal mixing period. Thus, the audio samples of the

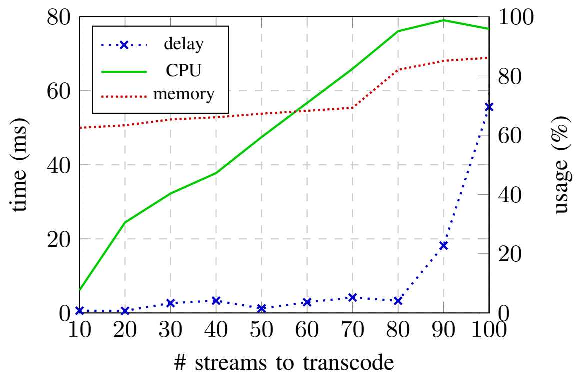


Figure 15: Delay and resources used in the relay when transcoding

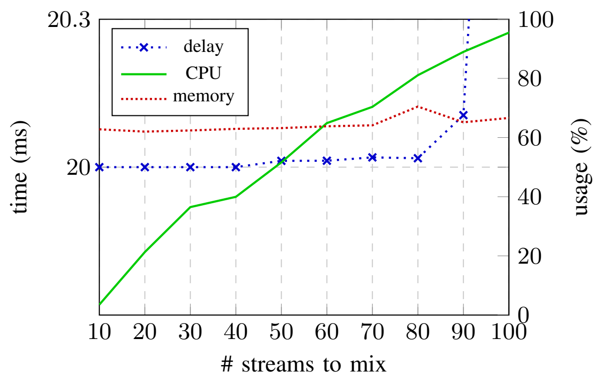


Figure 16: Delay and resources used in the relay when mixing

audio streams to be mixed are enqueued this time at maximum before being mixed. However, as the load supported by the relay increases, the actual mixing period diverges from the ideal, so the delay introduced by the mixing process increases. This effect is shown in Fig. 16. As can be seen, the relay can easily mix up to 40 audio streams. When a higher number of streams are mixed, since the mixing period is higher than the ideal, the extra delay is accumulated in every frame, so the end-to-end latency grows unacceptably.

In practice, multimedia activities usually include some floor-control mechanism or the participants use some codec voice activity detection to save network resources. Thus, the number of streams mixed or transcoded rarely reaches the maximum capacity of relays when deploying various relays in the overlay. In addition, the number of relays that a stream must traverse is limited to 2, so the delay introduced by transcoding and mixing operations is limited.

7. Conclusions

The resilience of an overlay network tailored to interactive multimedia communications has been improved using self-regulation. This overlay network is based on a mesh of RTP relays playing the role of reflectors, deployed in various multicast islands. The participants of a multimedia activity communicate with relays using IP multicast where available. The relays, which are organized in a full-mesh topology, forward media streams and are controlled by a Rendezvous Point. The overlay includes a self-stabilization technique to face relay failures, so the participants can be redirected to other relays using unicast.

A self-regulation scheme has been proposed to improve the performance of the self-stabilization technique, which enhances the resilience of the overlay network. This scheme is based on media transcoding and media mixing techniques, reducing the network bandwidth consumption when forwarding and receiving media streams of redirected participants. Several media transcoding and media mixing approaches have been analyzed using extensive simulations. Results show that the self-stabilization ability of the overlay can be significantly enhanced when implementing some of these approaches, leading to an improvement in the resilience of the overlay.

Full mixing is the best approach to improve the performance of self-stabilization when the ratio of participants per relay is high, since packet replication due to redirected participants decreases. In contrast, enhanced downstream transcoding is the most efficient approach when the ratio of participants per relay is low. In this scenario, redirected participants do not threaten the overlay performance as most of the bandwidth consumption is related to communications among the relays in the mesh. Furthermore, unlike the mixing approach, enhanced downstream transcoding is not affected by the location of data sources and always reduces the amount of traffic in the mesh.

Tests have also shown that the delay introduced by the transcoding and mixing techniques is limited, so the end-to-end latency observed by participants is acceptable for interactive multimedia communications.

Acknowledgment

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] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, J. Zahorjan, Detour: informed internet routing and transport, IEEE Micro 19 (1) (1999) 50–59.

[2] J. P. G. Sterbenz, D. Hutchison, E. K. Çetinkaya, A. Jabbar, J. P. Rohrer, M. Schöller, P. Smith, Resilience and survivability in communication networks: strategies, principles, and survey of disciplines, Computer Networks 54 (8) (2010) 1245–1265.

[3] D. Andersen, H. Balakrishnan, F. Kaashoek, R. Morris, Resilient overlay networks, in: Proc. of the 18th ACM Symp. on Operating System Principles, Chateau Lake Louise, Banff, Canada, 2001, pp. 131–145.

[4] P. Horn, Autonomic computing: IBM's Perspective on the State of Information Technology (2001).

[5] J. Li, P. L. Reiher, G. J. Popek, Resilient self-organizing overlay networks for security update delivery, IEEE Journal on Selected Areas in Communications 22 (1) (2004) 189–202.

[6] S. Birrer, F. E. Bustamante, A comparison of resilient overlay multicast approaches, IEEE Journal on Selected Areas in Communications 25 (9) (2007) 1695–1705.

[7] V. Venkataraman, K. Yoshida, P. Francis, Chunkyspread: Heterogeneous unstructured tree-based peer-to-peer multicast, in: Proc. of the 14th IEEE Int. Conf. on Network Protocols (ICNP 2006), Santa Barbara, CA, USA, 2006, pp. 2–11.

[8] J. Molleda, R. Usamentiaga, D. Garcia, F. Bulnes, Towards autonomic computing in machine vision applications. Techniques and strategies for a 3D reconstruction system in harsh industrial environments, in: Proc. SPIE on Image Processing: Machine Vision Applications IV, Vol. 7877, San Francisco, CA, USA, 2011, pp.–13.

[9] B. Li, K. Nahrstedt, A control-based middleware framework for quality of service adaptations, IEEE Journal of Selected Areas in Communication, Special Issue on Service Enabling Platforms 17 (9) (1999) 1632–1650.

[10] J. Xin, C.-W. Lin, M.-T. Sun, Digital video transcoding, Proceedings of the IEEE 93 (1) (2005) 84–97.

[11] P. V. Rangan, H. M. Vin, S. Ramanathan, Communication architectures and algorithms for media mixing in multimedia conferences, IEEE/ACM Transactions on Networking 1 (1) (1993) 20–30.

[12] R. Mohan, J. R. Smith, C.-S. Li, Adapting multimedia internet content for universal access, IEEE Transactions on Multimedia 1 (1) (1999) 104–114.

[13] S. Yang, S. Yu, J. Zhou, Q. Han, Multipoint communications with speech mixing over IP network, Computer Communications 25 (1) (2002) 46–55.

[14] J. O. Kephart, D. M. Chess, The vision of autonomic computing, Computer 36 (2003) 41–50.

[15] R. Sterritt, M. Parashar, H. Tianfield, R. Unland, A concise introduction to autonomic computing, Advanced Engineering Informatics 19 (3) (2005) 181–187.

[16] S. Dolev, Self-stabilization, MIT Press, Cambridge, MA, USA, 2000.

[17] M. Mamei, R. Menezes, R. Tolksdorf, F. Zambonelli, Case studies for self-organization in computer science, Journal of Systems Architecture 52 (2006) 443–460.

[18] A. Berns, S. Ghosh, Dissecting self-* properties, in: Proc. 3rd IEEE Int. Conf. on Self-Adaptive and Self-Organizing Systems (SASO 2009), Vol. 0, Los Alamitos, CA, USA, 2009, pp. 10–19.

[19] P. Nuño, J. C. Granda, F. J. Suárez, D. F. García, Self-* in multimedia communication overlays, Computer Communications 36 (7) (2013) 817–833.

[20] X. Zhang, J. Liu, B. Li, T.-S. P. Yum, CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming, in: Proc. of 24th IEEE INFOCOM, Miami, FL, USA, 2005, pp. 2102–2111.

[21] N.-F. Huang, Y.-J. Tzang, H.-Y. Chang, C.-W. Ho, Enhancing P2P overlay network architecture for live multimedia streaming, Information Sciences 180 (17) (2010) 3210–3231.

[22] M. Castro, P. Druschel, A. Kermarrec, A.-M.and Nandi, A. Rowstron, A. Singh, SplitStream: High-bandwidth multicast in cooperative environments, ACM Operating Systems Review 37 (5) (2003) 298–313.

[23] H. Kasai, M. Nilsson, T. Jebb, M. Whybray, H. Tominaga, The development of a multimedia transcoding system for mobile access to video conferencing, IEICE Transactions on Communications E85-B (10) (2002) 2171–2181.

[24] T. Huang, X. Yu, An adaptive mixing audio gateway in heterogeneous networks for ADMIRE system, in: Proc. 2nd Int. WS on Grid and Cooperative Computing (GCC '03), Shanghai, China, 2004, pp. 294–302.

[25] A. Trad, Q. Ni, H. Afifi, Adaptive VoIP transmission over heterogeneous wired/wireless networks., in: Proc. 2nd Int. WS on Multimedia Interactive Protocols and Systems (MIPS 2004), Grenoble, France, 2004, pp. 25–36.

[26] M. Craven, T. Le, P. Lardieri, DVoIP: Dynamic Voice-over-IP transformations for quality of service in bandwidth constrained environments, in: Proc. IEEE Military Communications Conference (MILCOM 2008), San Diego, CA, USA, 2008, pp. –6.

[27] I. Kouvelas, V. Hardman, J. Crowcroft, Network adaptive continuous-media applications through self organised transcoding, in: Proc. 8th Int. WS on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'98), Cambridge, UK, 1998, pp. –15.

[28] J. Guo, F. Chen, L. Bhuyan, R. Kumar, A cluster-based active router architecture supporting video/audio stream transcoding service, in: Proc. 17th IEEE Int. Parallel and Distributed Processing Symposium (IPDPS), Nice, France, 2003, pp. –8.

[29] J. Noh, B. Girod, P. Pogrzeba, S. Agarwal, J. Singh, K. Kim, Method and system for distributed audio transcoding in peer-to-peer systems, US Patent 8,392,201 (Mar. 5 2013), pp. –20.

[30] J. Brandt, V. Kahmann, L. Wolf, A flexible reflector for media streams, in: Proc. Kommunikation in Verteilten Systemen (KIVS), Vol. PI-61 of Lecture Notes in Informatics, Kaiserslautern, Germany, 2005, pp. 41–48.

[31] S. Mate, I. Curcio, Mobile and interactive social television, IEEE Communications Magazine 47 (12) (2009) 116–122.

[32] Y. Zhang, S. Mao, L. Yang, T. Chen, Broadband mobile multimedia: techniques and applications, wireless networks and mobile communications, Taylor & Francis, 2008.

[33] M. Radenkovic, C. Greenhalgh, Multi-party distributed audio service with TCP fairness, in: Proc. 10th ACM Int. Conf. on Multimedia (MULTIMEDIA '02), New York, NY, USA, 2002, pp. 11–20.

[34] B. Yu, K. Nahrstedt, A scalable overlay video mixing service model, in: Proc. 11th ACM Int. Conf. on Multimedia (MULTIMEDIA '03), New York, NY, USA, 2003, pp. 646–647.

[35] D. Khedher, R. Glitho, R. Dssouli, Media handling aspects of multimedia conferencing in broadband wireless Ad hoc networks, IEEE Network 20 (2) (2006) 42–49.

[36] X. Wu, K. Dhara, V. Krishnaswamy, Enhancing application-layer multicast for P2P conferencing, in: Proc 4th IEEE Consumer Communications and Networking Conference (CCNC 2007), Las Vegas, NV, USA, 2007, pp. 986–990.

[37] X. Gu, Z. Wen, P. S. Yu, Z.-Y. Shae, peerTalk: A peer-to-peer multiparty voice-over-IP system., IEEE Transactions on Parallel and Distributed Systems 19 (4) (2008) 515–528.

[38] C. D. Nguyen, F. Safaei, P. Boustead, Optimal assignment of distributed servers to virtual partitionsfor the provision of immersive voice communicationin massively multiplayer games, Computer Communications 29 (9) (2006) 1260–1270.

[39] X. Yu, Z. Yu, A distributed architecture of video conference using P2P technology, Journal of Networks 7 (11) (2012) 1852–1859.

[40] M. Aloqaily, F. Belqasmi, R. Glitho, A. Hammad, Multi-party/multimedia conferencing in mobile Ad hoc networks for improving communications between firefighters, in: Proc. 10th ACS/IEEE Int. Conf. on Computer Systems and Applications (AICCSA'2013), Fes, Morocco, 2013, pp. –7.

[41] J. C. Granda, P. Nuño, D. F. García, F. J. Suárez, Autonomic platform for synchronous e-training in dispersed organizations, Journal of Network and Systems Management (2013) –27.

[42] J. C. Granda, D. F. García, P. Nuño, F. J. Suárez, An efficient networking technique for synchronous e-learning platforms in corporate environments, Computer Communications 33 (14) (2010) 1752–1766.

[43] J. C. Granda, P. Nuño, D. F. García, F. J. Suárez, A self-managed resilient overlay network for interactive multimedia communications, in: Procs of 19th IEEE Symposium on Computers and Communications (ISCC2014), Madeira, Portugal, 2014, pp. 1–7.

[44] R. Prasad, H. Jamadagni, H. Shankar, Number of floors for a voice-only conference on packet networks - a conjecture, IEE Proceedings-Communications 151 (3) (2004) 287–291.