Voice over Application-Level Multicast

Nick Blundell, Norbert Egi, Laurent Mathy
Computing Department, Lancaster University, Lancaster, LA1 4WA, UK
{n.blundell, egi, laurent}@comp.lancs.ac.uk

Abstract

In this paper, we present a thorough and realistic analysis of voice (*i.e.* audio conferencing) over application-level multicast (ALM).

Through flexibility and ease-of-deployment, ALM is a compelling alternative group-communication technique to IP Multicast — which has yet to see wide-scale deployment in the Internet. However, proposed ALM techniques suffer from inherent latency inefficiencies, which we show, through realistic simulation and exploration of perceived quality in multi-party conversation, to be greatly problematic for the realisation of truly-scalable audio-conferencing systems over ALM.

By incorporating talkspurt data from a large and detailed corpus of multi-party conversation, and through using network-simulation techniques based on actual Internet latency measurements, we develop our previous work on the Application-Level Network Audio-Conferencing (ALNAC) routing protocol into a thorough analysis of the problem, leading to a novel model for assessing the perceptual quality of multi-party conversation and to novel techniques for speaker prediction. We show that through adaptation to conversational patterns, the ALNAC protocol can achieve perceptual quality for large-scale audio conferencing that, with little cost to each end-system node, is comparable to IP Multicast.

1 Introduction

It is well known that the mouth-to-ear latency of an echo-less voice-communication channel should not exceed 300 ms in order to allow natural conversation [2] — audible echo can reduce this threshold by two orders of magnitude.

This limit is especially important for Internet VoIP applications, where the communication channel comprises non-trivial application and network latency components. With typical one-way application latencies of 60–400 ms [6] and Internet round-trip latencies of 150–200 ms [4], such VoIP applications operate with communication-channel latencies that are at or above the upper threshold of human tolerance.

In particular, this poses a problem for application-level group communication techniques, which are inherently less latency- or cost-efficient than their scarcely-deployed network-level counterpart (*i.e.* IP multicast): for example, multiple unicasting between participants cannot scale to support even modestly sized groups; standard overlay-tree flooding (*i.e.* that is performed by proposed application-level multicast (ALM) techniques) results in highly-varied node-pair latencies; and centralised reflector servers do not accommodate well groups of widely distributed membership (*i.e.* since there is no obvious place to put the reflector). A specific solution is therefore required to support group audio applications.

In [1] we proposed ALNAC, a dynamic application-level multicast (ALM) routing protocol especially designed for audio-conferencing applications, and we argued that perceptual quality of multi-party conversation could be improved by exploiting the patterns in natural conversation that allow for prediction — with a high accuracy — of who will speak next in conversation.

In this paper, we develop the preliminary work in [1] into a thorough investigation of the problem and make the following contributions. In Section 2, we give an in-depth exploration of the specific effects of communication-channel latency on multi-party conversation, leading to a novel model of perceptual quality. In Section 4, we propose, and conduct a thorough analysis of, a next-speaker prediction algorithm, using a large corpus of highly-detailed talkspurt data from actual multi-party conversation. Finally, in Section 5, we evaluate, by simulation, our ALM-based audio-conferencing proposal under conditions of realistic network latency and through using a realistic model of multi-party conversation.

2 Issues of Latency in Multi-Party Conversation

In interactive scenarios, latency is a problem usually because it *cannot* be perceived: in fact, only when a source's sound is reflected (echoed) back can latency be gauged; otherwise, the listener's brain interprets what is heard or what is not heard as events that happen in real-time, for example: we would quite-happily perceive a live radio show as such despite that it may in fact have a two-minute censorship delay.

Thus, when engaged in conversation we are constantly (subconsciously) projecting times at which *responses* to our spoken *cues* should arrive; if they do not arrive within our expected time range (a range bounded by well-studied latency-tolerance threshold), we perceive that they will never arrive and repeat cues unnecessarily in an attempt to repair the conversation.

With these two considerations in mind, we can argue, therefore, that perceived quality is not simply dependent upon a communication channel's latency but upon the delay with which specific responses are heard after their cues. This observation is particularly relevant to multi-party conversation, since not only will a participant hear responses to their own cues — if they choose to speak — but they will also hear responses to the cues of other participants, and so perceptual quality of those responses will be dependent not upon the *absolute latencies* with which they are heard but upon the *difference* in absolute latencies of those responses and their cues.

2.1 Definition of Cues and Responses

We define a *cue* to be an act of speech that beckons an immediate *response*, such that a listener's perception of quality will be degraded if sufficient delay occurs between a cue and its response.

We therefore make a distinction between those speech acts that *do* cue a response and those that *do not*, for example: if a participant continues to talk, a listener will expect the talkspurts to follow one another, with one talkspurt beckoning on the next; likewise, if two participants are engaged in conversation, one participant will expect a prompt reply from the other upon completion of their turn; however, in situations where no (immediate) response is expected, such as at the end of the discussion on a specific topic, or upon a participant posing a rhetorical question, we say that the next talkspurt was not *cued* by the previous.

Our observations of semantic information (*e.g.* transcribed speech, and adjacency-pair tagging) and analysis of talkspurt data in meetings of the ICSI corpus (see Section 4) show that such cue–response relationships between pairs of talkspurts can be easily distinguished by the length of gaps (silences) that separate talkspurts. The result of this experiment upholds observations of the study of linguistics that conversational turns are typically delimited by etiquette silences of no more than 1 second [11].

We therefore use a simple heuristic to determine whether a talkspurt is a response or not: if the talkspurt occurs less than 1 second after another — or, indeed, if it overlaps with another talkspurt — we define it as a response. Upon identifying a response talkspurt, we then scan backwards to determine which talkspurt, if there is more than one possibility, cued the response, selecting as the cue the most-recent talkspurt generated by the participant who spoke for the highest proportion of time in the past 2 seconds; examination of semantic information in the corpus data shows that this heuristic is accurate in identifying cues that have multiple responses (*e.g.* when two or more people collide to answer the same question or acknowledge their understanding).

2.2 Issues of Stream Synchronisation

Due to the inherent latency inefficiency of ALM techniques, there is a potential that participants of a multi-party communication system will observe highly-varied network delays between streams, which, by affecting the synchronisation of responses and their cues, will impact upon perceived quality. (As an exaggerated example, consider how quality might be perceived by a participant who hears the answer to a question before hearing the question itself.)

Since to the authors' knowledge there have been no studies on the effects of stream desynchronisation on the perceived quality of multi-party conversation (*i.e.* from a listener's perspective), we performed a simple listening experiment in which we purposely desynchronised a participant channel of a recorded meeting from the ICSI meeting corpus: in the experiment, we took one recorded audio channel (a channel of a participant who was engaged in conversation for a large proportion of the particular meeting segment) and shifted it by various time constants, before mixing all of the separate channels into a single audio file; a set of mixes were thus created (including the original mix without shifting), and six volunteers, who had no insight into the particular transformation that was performed, were asked to categorise between those mixes that sounded strange and those that sounded normal.

Interestingly, we found that none of the listeners in the experiment could perceive a difference between mixes that were desynchronised by less than or equal to 1,000 ms, which indicates that we have a higher tolerance to the lateness of responses when we listen to a conversation than when we are actively engaged in it (in which case the maximum mouth-to-ear round-trip tolerance is about 600 ms); for desynchronisation of over 1,000 ms, it became obvious to the listener that some transformation had been performed, since the etiquette silences that delimited turns were mostly annihilated and, with further desynchronisation, speech would overlap unnaturally on turn boundaries (*i.e.* before one participant finished speaking, another would start with a response).

We concede that this is by no means an extensive study of the problem of desynchronisation in multi-party conversation, but we postulate that this relaxed tolerance is apparent in our results since we pay less attention (subconsciously) to the timing of responses when we are listening to the consecutive turns of other participants than when we are actively seeking to take a turn ourselves.

2.3 Quality Model for Multi-Party Conversation

In line with our observations on quality-perception in multi-party conversation, we propose a perceptual-quality model that is not based on channel latency, as has so far been considered in the literature, but rather on the 'lateness' of individual spoken responses with respect to their cues.

In [2], the authors proposed a simple utility function for describing the perceived quality of mouth-to-ear channel latency in which two score-levels are defined: a high score level to reflect 'very good' latency perception, and a low score level to reflect 'bad' latency perception. We base our own utility function on a similar principle but instead consider tolerance to round-trip mouth-to-ear latency, since conversation — as a two-way process — is affected by round-trip latency and since asymmetric latencies are highly-likely in ALM. In addition to the original utility function, based on the results of our experiment with stream desynchronisation (see Section 2.2), we extend the function to distinguish between the tolerance thresholds to response lateness for a participant's own cues and for the cues of other participants.

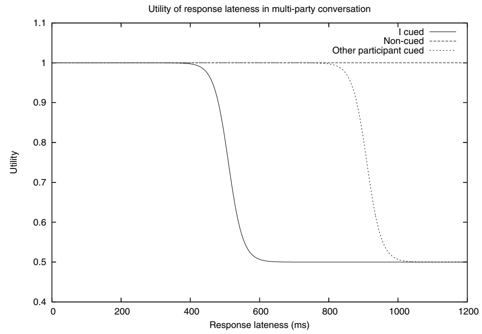


Figure 1: Utility function for the perceptual quality of response lateness in multi-party conversation.

The resulting utility function for perceptual quality is depicted in Figure 1. Note that, in the model, since a non-cued talkspurt cannot be perceived as being late, it is automatically awarded a score of 1.

3 ALNAC: A Dynamic Overlay Routing Protocol

ALNAC (Application-Level Network Audio-Conferencing routing protocol) [1] is a light-weight ALM routing protocol, designed especially to optimise audio-packet delivery for those audio-conference participants who are most sensitive to communication-channel latency (*i.e.* those who are currently engaged in conversation), whilst minimising the impact of such optimisation on members that are least sensitive to communication-channel latency (*i.e.* those members that take only a listening role in the current conversation).

More precisely, ALNAC operates over an ALM tree structure. ALNAC adapts a basic flooding technique whereby a speaker sends audio samples to the tree root and to its children (for forwarding in their respective sub-trees). The adaptation is that a speaker will send audio samples, in addition to the root, directly to a set of predicted next speakers who are identified as highly likely active participants in the current conversation by a prediction algorithm. On the other hand, because the out-degree of a node (i.e. the maximum number of forwarding the node will do) can be limited due to bandwidth constraints, some of the speaker's children on the ALM tree may have to be *deprived* from receiving the audio samples from the speaker directly. To ensure that all samples are eventually flooded to all nodes in the tree, a speaker will *delegate* the responsibility for supplying the deprived nodes among the nodes to whom it *is* sending directly. Note however, that delegation can be recursive (i.e. a supplier can further delegate). Figure 2 illustrates the audio sample distribution process in ALNAC.

We therefore see that, in essence, ALNAC builds a dynamic overlay over an ALM tree (as opposed to adapting the tree to conversation changes).

Note that in [1], a very primitive, static prediction algorithm was proposed. A more efficient and adaptive algorithm is described in the following section.

4 Next-Speaker Prediction

In [1], through the analysis of a limited number of textual transcripts of actual conversation and packet-trace files of an audio-conferencing application, we showed that in natural, multi-party conversation there is a high correlation between those participants who spoke recently and those who will speak next; the explanation for this result lies in the relationships be-

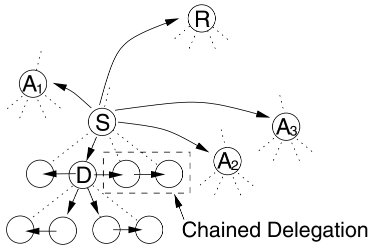


Figure 2: Dynamic routing of ALNAC through the process of delegation.

tween conversational turns, such as *adjacency pairs* (*e.g.* questions and answers, exclamations and responses, *etc.*), which have been well-documented in the study of conversation analysis [10].

In the context of audio conferencing over application-level multicast, we define the problem of next-speaker prediction as a problem of maximising the probability that one participant of a constrained set of recent-speaking participants, which set we refer to as the *backlog*, will speak next. Thus, the role of a next-speaker prediction algorithm essentially is to create a prioritised list of participants, ranking them by their level of 'activeness' in the conversation, such that a minimum backlog may be extrapolated from the priority list to perform optimised overlay routing.

In this section, we extend our previous work on next-speaker prediction into a more-complete analysis through the incorporation of corpus data collected and processed by the ICSI meeting project [5]. The corpus comprises the data of over seventy full-length meetings of natural, multi-party conversation, featuring interactions among wide varieties of participants (*i.e.* gender, age, ethnicity, *etc.*), and was produced primarily to aid linguistical research on group conversation and interaction. The data for each meeting comprises recordings of per-participant audio and highly-detailed transcripts, painstakingly annotated per-talkspurt with timing and semantic information. Figure 3 shows a sample of talkspurt patterns plotted from corpus meeting.

Using only timing information of talkspurts, as is readily available with little processing overhead to participants of an audio conference, we present one of our next-speaker prediction algorithms, called the *turn based* algorithm, that is heuristically derived through talkspurt analysis of the ICSI meeting corpus data and is easily implementable, but provides high performance in next-speaker prediction; finally, we give an evaluation of the algorithm.

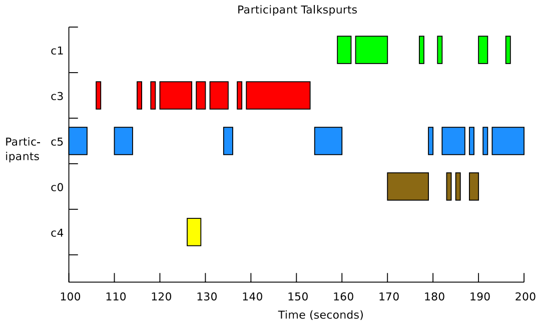


Figure 3: A sample of talkspurt activity among participants of the ICSI meeting corpus.

4.1 Prediction Algorithm

Our *turn based* next-speaker prediction algorithm presented in this section follows a strategy of associating with each group member (*i.e.* conference participant) a *priority* which quantifies the recent conversational contribution of the participant (and thus, the participant's likely immediate future contribution). The algorithm takes as input (a description of) audio samples/packets and produce a list of participants (ordered according to their computed priorities) on detection of turn bounderies (*i.e.* at points of speaker-change) to reflect changing levels of participation throughout the course of the meeting.

An issue for next-speaker prediction are short turns representing the natural sounds and speech events, such as unintentional talkspurts (*e.g.* coughing, laughing, and environmental noises) and intentional back-channel talkspurts (*e.g.* 'yeah', 'hmmm'), which can bias prediction by falsely indicating that a particular participant is currently engaged in conversation. Analysis of the ICSI corpus shows that disregarding turns shorter than 800 ms alleviate this problem.

The reader should note that each participant runs an independent instance of the next-speaker prediction algorithm and that a description of all audio packets (both received and produced by a participant) are used as algorithm input.

Initially, the algorithm assigns a priority of 1.0 to each participant. In order to remember recent levels of participation, the priority of each participant is adjusted using the following low-pass filter

$$pri\_P(t) = \alpha * pri\_P(t-1) + \beta * isTurnOf(P) \quad (1)$$

where pri_P is the priority value of participant P, α and β are decrement and increment factors respectively, and $isTurnOf(n)$ evaluates to one if the previous turn (the one that just finished) belongs to participant n and zero otherwise.

We fix the value of β to 1.0 and use α as a parameter to tune responsiveness of the algorithm, for example: if α is small, the algorithm will 'forget' faster the recent activity of a participant; and if α is larger, more weighting will be given to those participants with long-term activity.

Through heuristical evaluation of the corpus data, we find that $0.5 \leq \alpha \leq 0.9$ is a suitable range for maximising prediction under a range of conversational patterns.

Our next-speaker prediction algorithm is conceptually simple and easy to implement. However, in order to achieve effective ALNAC routing in all circumstances, the following simple extensions are proposed.

In order to keep actual backlog sizes (and therefore delegation) to a minimum, while still achieving high prediction accuracy, we introduce the concept of *backlog priority threshold*, whereby prediction algorithms will only return a priority-ordered list of participants whose priority is higher than the backlog priority threshold. Obviously, a higher threshold forces the algorithms to 'hide' participants whose recent conversational contribution is 'minor'.

Since the analysis of the prediction algorithm showed some variety in accuracy for different conversational circumstances under a range of values for the respective 'responsiveness' parameters (α), we propose a simple self-tuning mechanism based on the feedback of prediction accuracy to improve next-speaker prediction under circumstances of generic conversation (*i.e.* long/short-lived discussions, conversational topic changes, *etc.*) that encompass those observed in data of the ICSI meeting corpus. The self-tuning technique operates as follows: we express the value of the algorithm's 'responsiveness' parameter as a percentage of its optimal range (*i.e.* $0.5 \leq \alpha \leq 0.9$), such that 0% represents the least-responsive setting (*i.e.* $\alpha = 0.9$) and 100% represents the most-responsive setting (*i.e.* $\alpha = 0.5$). We initialise this parameter at 0% (*i.e.* least sensitive) and adjust it as follows: when a correct prediction is made, the parameter is lowered by a decrement percentage, *decr*%, and when an incorrect prediction is made, the parameter is increased by an increment percentage, *incr*%, such that it cannot rise above 100% or fall below 0%.

Through analysis of the algorithm against the complete meeting set of the ICSI corpus data, we find that prediction is optimised if the algorithm's responsiveness is increased slowly ($20 \leq incr \leq 25$) after an incorrect prediction and decreased quickly ($50 \leq decr \leq 100$) after a correct prediction.

4.2 Evaluation of the Algorithm

In Figure 4 the results of backlog priority threshold analysis show the prediction accuracy of the prediction algorithm against average backlog size. Note that, the curve labelled 'Initial algorithm' represents the prediction accuracy of our rudimentary algorithm from [1], in which prioritization of participants for prediction is based upon only the order of recent talkspurts.

In Figure 4, we see that the turn-based algorithm gives improved performance over the initial algorithm; this occurs as a result of the algorithm being capable of making intelligent judgements as to whether a talkspurt is significant in prediction or not, for example: whether a talkspurt is a short burst of noise or backchannel speech (*i.e.* 'mmm-hmm', 'yeah', *etc.*) from participants who have no intent to become engaged in the current conversation.

In summary, we see that a high prediction accuracy can be achieved in multi-party conversation by considering only a small backlog of previous speakers (≤ 3); this result confirms the results of our less-extensive analysis of textual transcript turn patterns and packet traces in [1].

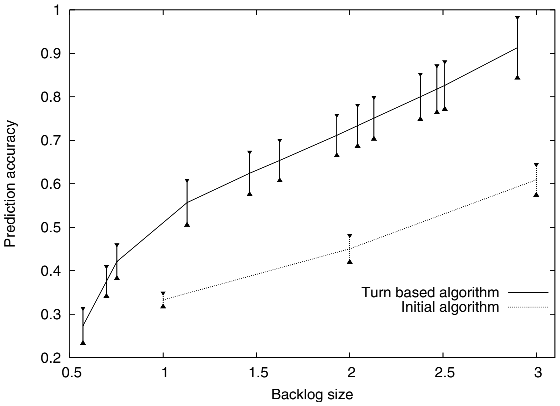


Figure 4: Comparison of the algorithms' prediction efficiency

5 Simulations

Since existing topology-based simulators cannot simulate the realistic dynamics of Internet node-pair latencies (*i.e.* due to complexities of traffic patterns and network structuring), we implemented an event-based network simulator that uses latency matrices, populated by actual Internet latency measurements. The latency matrices were obtained for 1740 arbitrary

Internet hosts from [3] and for PlanetLab [9] nodes from [12].

As the basis for ALM, we implemented the Tree-Building Control Protocol (TBCP) [8], a low-overhead control protocol for rapidly building latency-optimised, cost-constrained overlay trees among groups of network nodes.

Figures 5 and 6 are presented here to give the reader a feel for the experience of individual participants over the duration of a single audio conference.

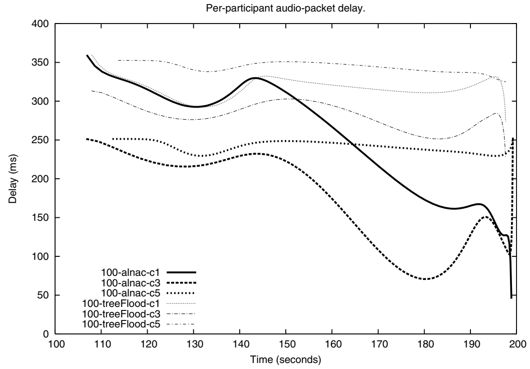
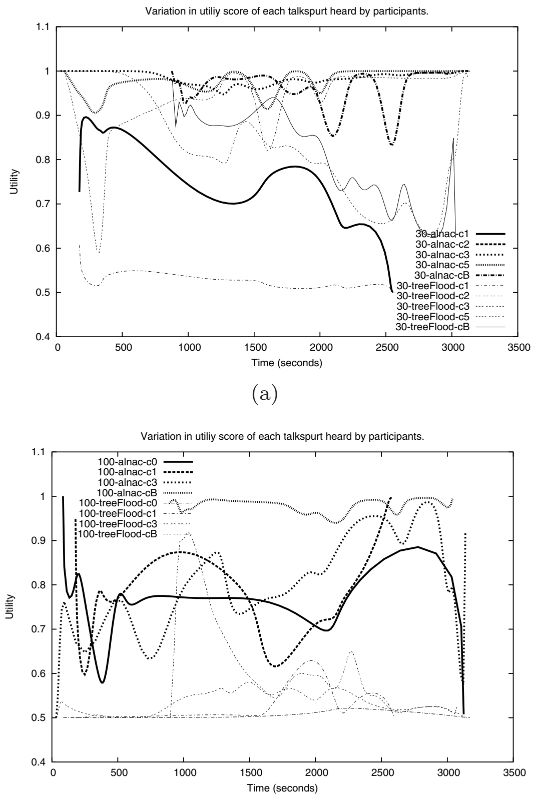


Figure 5: Variation in audio-packet delay over time for a selection of participants.

Figure 5 shows the trend in audio-packet delay that was experienced by a selection of participants ('c1', 'c3', and 'c5') in the time range 100–200 seconds of meeting 'Bmr024' from the ICSI meeting corpus, simulated using a 100-node overlay tree. General variations in delay may be accounted for by changes over time in which participant is currently speaking, however we can make the first observation that when using ALNAC, the network delay experienced by participants 'c3' and 'c5' was consistently lower than when non-adaptive tree-flooding was used; and upon examining the talkspurt patterns for this segment of the meeting (see Figure 3), we see that those two participants appear to have been talking with each other, and so ALNAC appears to have adapted their network delays accordingly.

The next interesting feature of this figure is the 'forking-point' in the delay trends of tree-flooding and ALNAC protocols for participant 'c1' that occured at time ∼140 seconds; by looking again at the talkspurt patterns in Figure 3, we see that this point signifies how at ∼170 seconds ALNAC had adapted overlay routing such that particiapant 'c1' began to receive audio-packets with a lowered delay *immediately* as he/she became engaged in conversation.

Figures 6(a) and 6(b) show, respectively, the variation in utility (a perceptual-quality function based on our model in Section 2.3) of responses heard by a selection of participants in a 30-node and 100-node audio conference. As a guide, a utility score close to 1.0 should be interpreted as 'very good' and a score close to 0.5 as 'bad', and a score between 0.5 and 1.0 should be interpreted as 'good'.



(a)

(b)

Figure 6: Variation in the QoS utility of talkspurts heard by a selection of participants.

In Figure 6(a), we see that, with the exception of participant 'c1', the selected participants experienced good to very-good quality when non-adaptive tree-flooding and ALNAC were used; for non-adaptive tree-flooding, this shows us that those participants have been fortunate in their location in the overlay tree; however, throughout the audio conference, participant 'c1' appears to have been 'unfortunate' in its tree location, and as a result 'c1' experiences bad quality for tree-flooding. In general, we see that, through adaptation to conversation patterns, ALNAC tends to keep a participant's perceived quality closer to 'very good' than does non-adaptive tree-flooding.

It should also be noted here that even with typical unicast latencies, VoIP applications will struggle to achieve a utility score of 1.0 (*i.e.* a round-trip mouth-to-ear latency of ≤ 300 ms).

In Figure 6(b), we see that perceived quality among participants is greatly affected by the large membership size when non-adaptive tree-flooding is used, with the perceived quality of *all* the selected participants dropping to 'very bad' at some point of the audio conference. This figure may best be interpreted by a comparison to stalactite and stalagmite structures, where ALNAC maintains perceptual quality at or below the 'very good'-quality ceiling, whilst non-adaptive tree-flooding results in perceptual quality that occasionally rises above the 'bad'-quality floor.

6 Conclusions

In this paper, we have presented a novel and thorough investigation of two properties of multi-party conversation that are highly important in the realisation of VoIP applications over ALM: (i) the effects of communication-channel latency on quality perception in multi-party conversation; and (ii) the problem of next-speaker prediction in multi-party conversation (*i.e.* which participants are currently most-sensitive to communication-channel latency). The two main contributions, namely our quality model for multi-party conversation and our efficient next-speaker prediction algorithms, although developed in the context of our work on ALNAC, are readily applicable in the wider context of audio-conferencing systems. Indeed, they can, for instance, be used to evaluate and guide the operation of related proposals such as ACTIVE[7] (a proposal based on the strategy of shaping the ALM tree so that active speakers are near the root).

We have also presented the ALNAC protocol and conducted simulations that model, realistically, characteristics both of the network and of multi-party conversation.

Based on our analysis and simulations, we conclude that in order to support truly-scalable audio conferencing over ALM, an ALM routing protocol *must* be reactive to the conversational patterns of participants, such that perceived quality may be improved for not just *some* of the participants (*i.e.* by fortune of their location in the overlay tree(s)) but for *all* participants. The ALNAC protocol, including its next-speaker prediction algorithm, was shown to be a scalable, elegant and general solution to this problem, capable of efficiently supporting both meeting-type and orator-type audio conference applications.

References

[1] Blundell, N., Mathy, L.: Minimising *Perceived* Latency in Audio-Conferencing Systems over Application-Level Multicast. In Proceedings of MIPS 2004, Grenoble, France, Nov 2004.

[2] Boutremans, C.,Le Boudec, J.-Y.: Adaptive Delay Aware Error Control For Internet Telephony. In Proceedings of the 2nd IP-Telephony Workshop, New York, April 2001.

[3] Gil, T. M., Kaashoek, F., Li, J., Morris, R., Stribling, J.: King Dataset. http://www.pdos.lcs.mit.edu/p2psim/kingdata/, August 2004.

[4] Gummadi, K.P., Gummadi, R., Gribble, S. D., et al.: The Impact of DHT Routing Geometry on Resilience and Proximity. In Proceedings of the ACM SIGCOMM 2003, Karlsruhe, Germany, August 2003.

[5] Janin, A., Ang, J., Bhagat, S., et al.: The ICSI Meeting Project: Resources and Research. Proceedings of NIST ICASSP 2004 Meeting Recognition Workshop, Montreal, May 2004.

[6] Jiang, W., Koguchi, K., Schulzrinne, H.: QoS Evaluation of VoIP End-Points. Proceedings of IEEE International Conference on Communications (ICC 2003), Anchorage, Alaska, May 2003.

[7] Liu, L., Zimmermann, R.: ACTIVE: Adaptive Low-Latency Peer-to-Peer Streaming. In Proceedings of the Twelfth Annual Multimedia Computing and Networking (MMCN '05), San Jose, California, January 2005.

[8] Mathy, L., Canonico, R., Hutchison, D.: An Overlay Tree Building Control Protocol. Proc. of Intl. workshop on Networked Group Communication (NGC), Nov 2001. 76–87

[9] PlanetLab. http://www.planet-lab.org.

[10] Lectures on Conversation. Blackwell, Oxford, UK, 1992.

[11] Schmitz, U.: Eloquent Silence. Linguistik-Server Essen (LINSE), 1994.

[12] Stribling J.: PlanetLab all pairs ping data. http://pdos.csail.mit.edu/~strib/pl_app/.

[13] Chu Y.-H., Rao S. G.,Zhang H.: A Case for End-System Multicast. In Proceedings of ACM SIGMETRICS 2000, Santa Clara, California, US, June 2000.