Successful Multiparty Audio Communication over the Internet

Vicky Hardman, M. Angela Sasse and Isidor Kouvelas

Department of Computer Science

University College London (UCL)

London, UK

Communications of the ACM

May 1998

Outline

- Introduction
- Problems with Current Tools
- The Robust Audio Tool
- Conclusions

Introduction

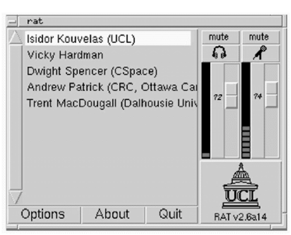
- Internet has point-to-point (unicast)
- Web allows download and playout of MM
- Multicast conferencing requires
  - Real-time interactive audio and video
  - Many participants at once
- Of audio and video, audio most important
  - Get it right, first

Introduction

- Mbone provides test bed for diverse apps
  - Teleconferencing
  - Telepresence
  - Distance education
- Multicast audio quality hurt by
  - Packet loss
  - Lack of real-time operating system support

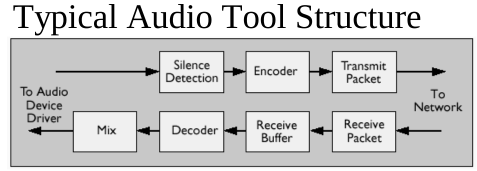
Introduction

- Many publicly available multicast audio tools
  - Visual Audio Tool



- Power meters to indicate send-receive volumes (why?)
- Highlight speaker (why?)

Typical Audio Tool Structure



- Sample 20, 40 or 80 ms
  - (Tradeoff between sample size and quality?)
  - (Tradeoff between sample size and network load?)
- Use silence detection
  - Measure of energy
  - Bandwidth savings up to 50% (2-way)
  - Instead, use “push-to-talk” button (why?)
- (What is “Mix”?)
- Buffer to remove some jitter

Problems with Existing Tools

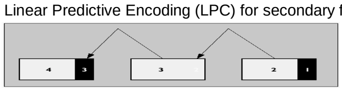
- Delay not usually a big problem
  - Typically under 400-600 ms
- Loss can be problematic
  - 10% is max tolerable with silence substitution
  - International links typically have 20-25% loss
- No real-time OS support
  - Processes to send-receive may not be scheduled in time, leading to gaps in playout

Acoustic Problems with Existing Tools

- Echoes and feedback
  - Push-to-talk helps, but not silence detection
- Silence detection
  - Clips words, may pick up background noise
- Lack of distance cues
  - Causes mismatch of volume
- Restricted intelligibility
  - Caused by mere toll quality audio (?)
- Monaural sound
  - Hard to identify speakers

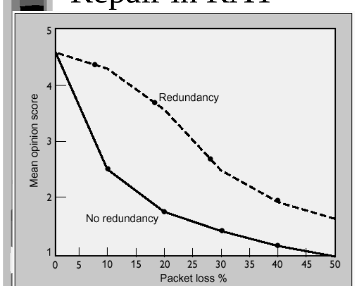
The Robust-Audio Tool

- Has ‘hands-free’ design
  - Speech detection
- Workstation scheduling features (we’ll cover)
- Loss seen as most significant
  - Uses redundancy to repair
    - + Linear Predictive Encoding (LPC) for secondary frame



  - Multiple loss repaired through waveform substitution on second packet
    - + Brain can smooth over

Repair in RAT

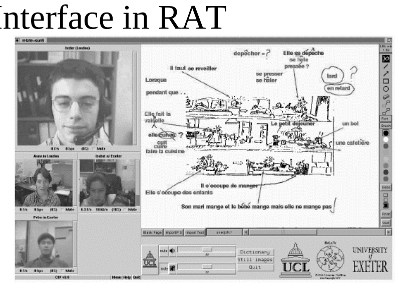


- Up to 20% loss ok
  - 30% with waveform (not shown)

Silence Detection in RAT

- Uses simple energy measure
  - Supplement with ‘rule-based’ approach(?)
- Adaptive silence threshold
  - For changing background noise
- Works so well, do not need ‘push-to-talk’ feature

Interface in RAT



- Video and shared whiteboard
- Audio and video separate, so need synch
- Power-meter for volume

Future Work?

Conclusions and Future Work

- Multicast has potential, but problems to fix
- RAT fixes loss, has improvements to interface
- Loss now ok, acoustic problems next
  - (A previous paper demonstrated this)
- Need to have responsive protocols
  - (Our previous topic)
  - Use alternate encoding to reduce rate
- Network may use priority queuing to help
  - (Upcoming topic)