--

--

A Cooperative Packet Recovery Protocol for Multicast Video

N. F. Maxemchuk
AT&T Labs - Research
Florham Park, N.J.

K. Padmanabhan
Fujitsu Labs

S. Lo
Lucent Tech - Bell Labs
Murray Hill, N.J.

DIFFERENCE IN REQUIREMENTS

- **DATA:**
  — Near Perfect
  — Better Off Late Than Wrong

- **REAL-TIME AUDIO or VIDEO**
  — News, Sporting Events, Conference Sessions, Courses
  — Natural Redundancy Permits Intelligibility With Errors
  — A Few Second Delay Is Permissible
  Even With Questions
  — Lost Phrases Cannot Be Recovered

- **INTERACTIVE AUDIO or VIDEO**
  — Short Delay To Maintain Conversation
  — Garbled Phrases Or Lost Seconds Can Be Recovered
  What did you say?
  Preferable to Continuous Noise

1243

A MULTICAST PROTOCOL
FOR REAL TIME VIDEO OR AUDIO

1. OBSERVATION

— Mbone Video Retransmission Is Within The Speed And Memory Capabilities of Current PC's

— Transmission Rate = 128 Kbps
128 KBytes of Memory = 8 Seconds
Enough for 2 or 3 Retrys, Most of the Time

2. DIFFERENCES BETWEEN VIDEO OR AUDIO AND DATA

— Continuous Transmission *VRS* Stop And Wait
NACK Instead of ACK

— Inherent Redundancy Makes Imperfection Acceptable
Stop Retrying After Small Number of Attempts
Retransmit Server and Source Not Colocated

UNIQUE CHARACTERISTICS

— Implemented Without The Cooperation of the Source or Receivers
Improve the Quality of Existing Mbone Broadcasts

— Improves Quality For Some, Not All, Receivers
Provides a Service That An ISP Can Sell

— Fewer Active Participants
Simpler Than Most Reliable Multicast Protocols

— Network Variables Are Learned Rather Than Predicted
It Is Almost As Good to Be Right From Now On
As To Have Always Been Right

DESCRIPTION OF THE PROTOCOL

THE PLAYERS

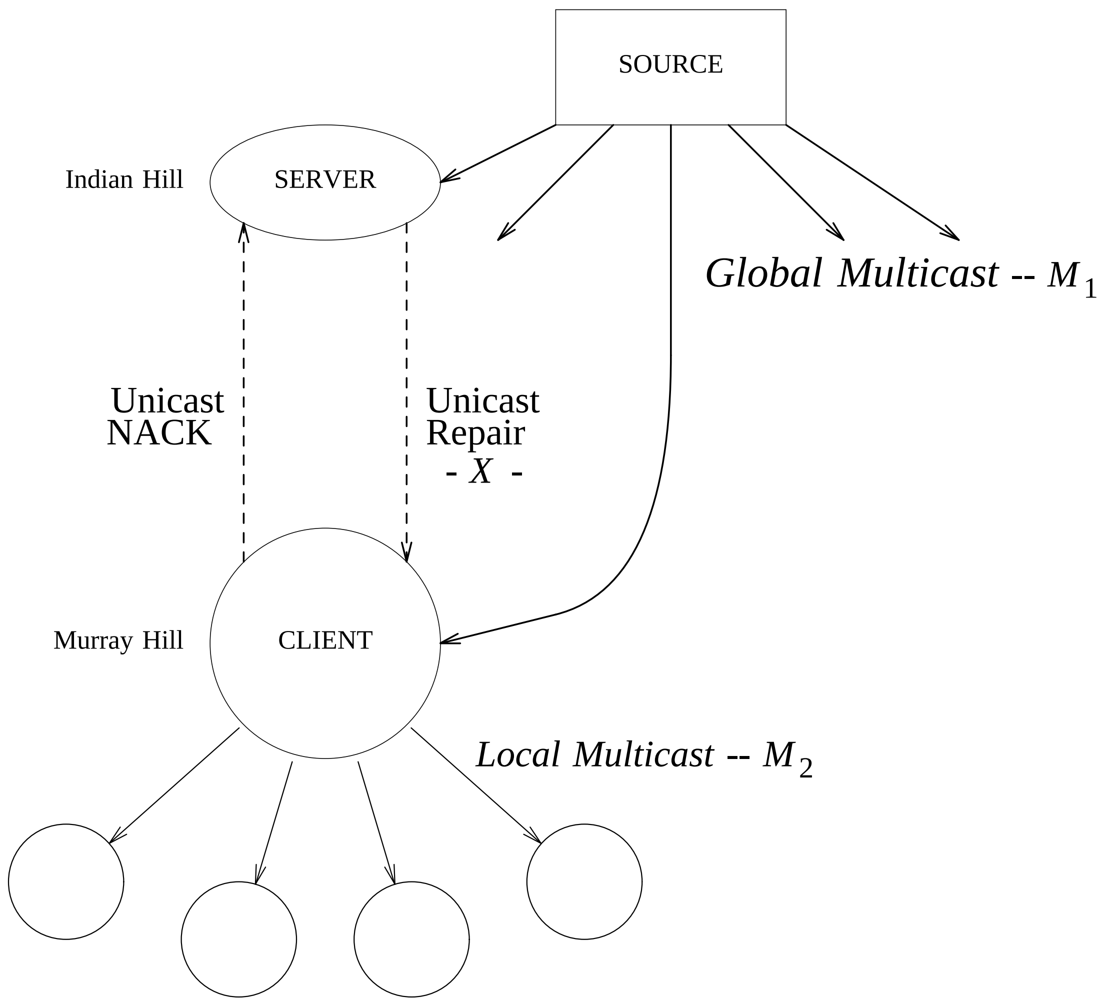
1. **THE SOURCE**
   — Multicast Packets (*Including Sequence Numbers - RTP*)

2. **THE RETRANSMIT SERVER** *(Near the Source)*
   — Receives Multicast Packets From The Source
   More Than Distant Receiver
   — Retransmits Packets When A NACK Is Received

3. **THE RETRANSMIT CLIENT**
   — Receives Multicast Packets From The Source
   — Requests Missing Packets From The Retransmit Server
   Based on Sequence Numbers
   — Multicasts Repaired Sequence
   After K Seconds

4. **THE RECEIVERS** *(Near the Retransmit Client)*
   — Non-Participants - Receive Multicast From The Source
   — Participants - Receive Repaired Multicast

EXPERIMENTAL CONFIGURATION



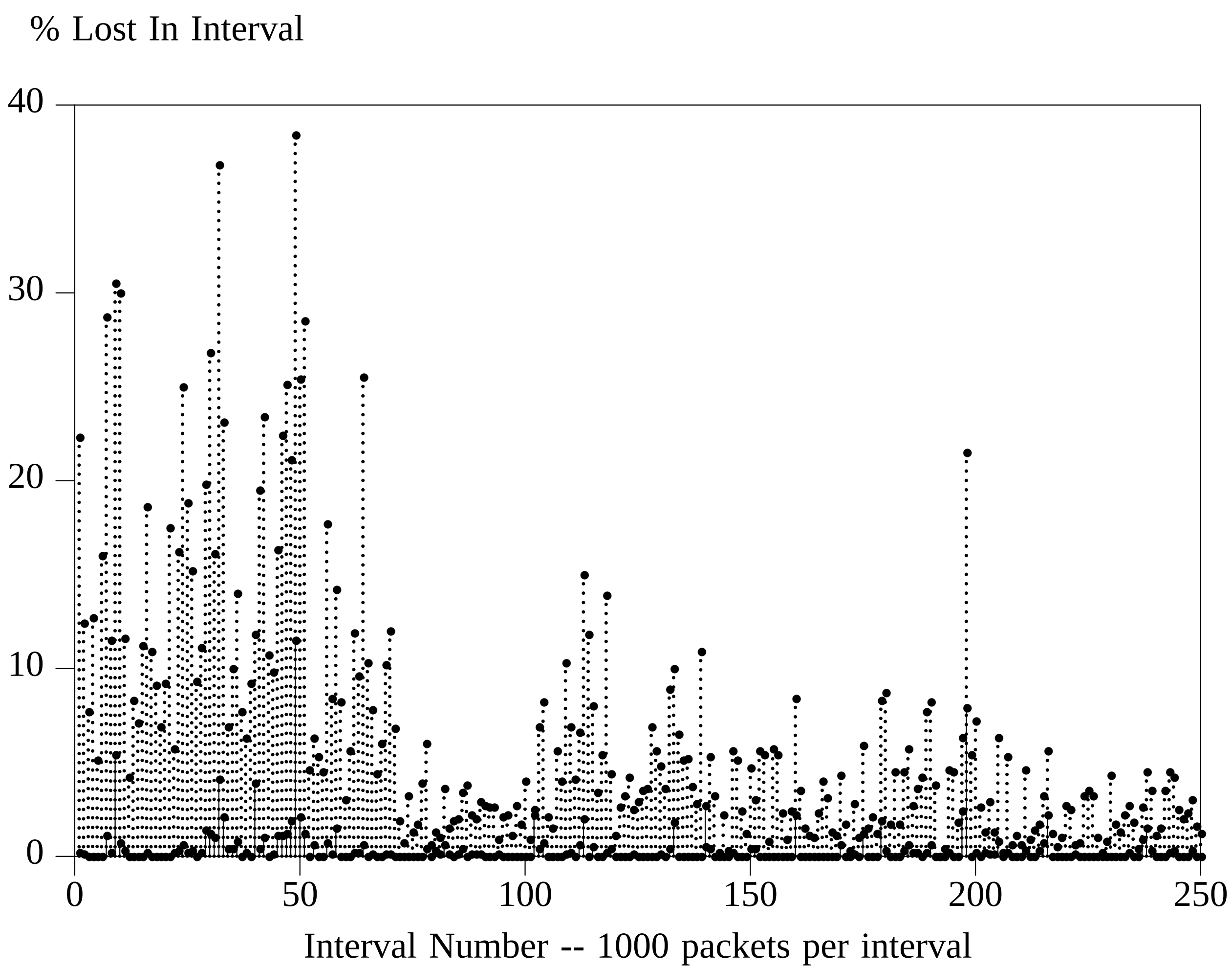
1199

TYPICAL SESSION

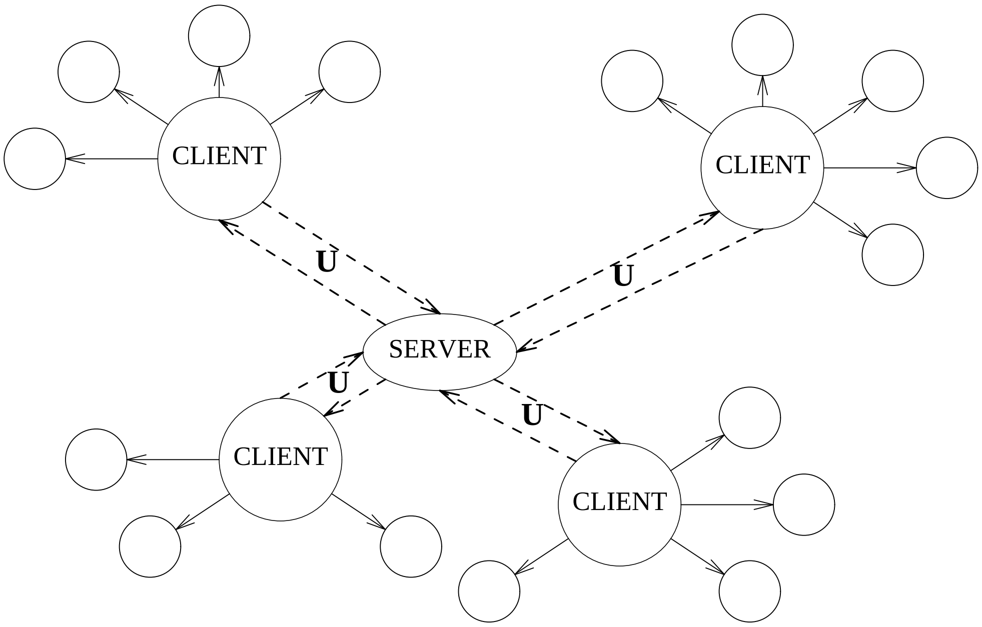
2 ADJACENT RECEIVERS

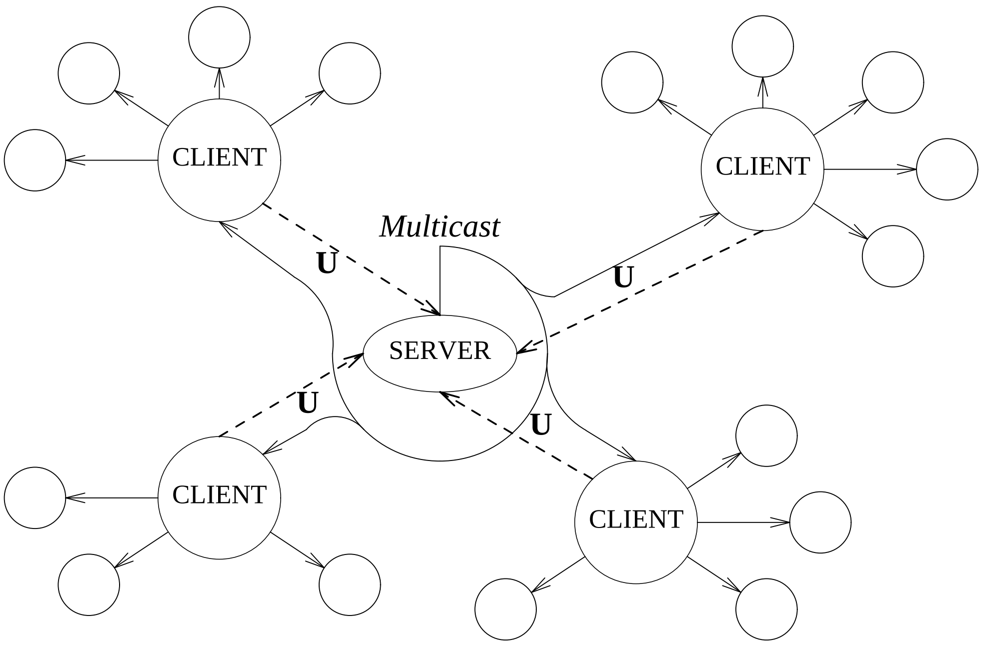
"." Receives Raw Multicast

"." Receives Corrected Multicast



GENERALIZATION OF EXPERIMENT





"IMPROVEMENTS" ON MULTICAST RECOVERY

NACK Supression:

Problem: When A Packet is Lost On a Multicast Tree, Everyone Downstream Misses the Packet and Sends A NACK.

A Solution: Send A NACK Immediately With Probability *P*. Send a NACK After Time *T* If It Is Still Needed

Multicast Retransmission Explosions:

Problem: When Retransmissions Are Multicast, And the Number of Receivers is Large, Most Packets Are Missed By Someone And Re-Multicast

A Solution: A Layered Architecture of Retransmit Servers. Send and Service Requests on Sub-trees.

Observations:

These Problems Are Fun to Solve, But Have Not Been Needed.

The Problems Are Caused By Large Numbers Of Protocol Players, But The Retransmit Agents Reduces The Number Of Players

VARIABLE SYSTEM PARAMETERS

Problem: Delay And Loss On The Internet Are Difficult To Estimate, And May Change During A Session.

The protocol depends on events, such as retrys, occuring after a round trip delay.

Solution: Operate With Reasonable Parameters. If They Are Wrong, Change Them.

Incorrect parameters waste resources or cause distortion. The cost of learning rather than *accurately* predicting the parameters are the resources that are lost prior to correction. For "long lived" sessions this should be a small cost.

Examples:

Time To Retry Retransmission Request: If the response to an earlier request is received after you retry, you did not wait long enough

Replay Delay At The Retransmit Client: If a packet is received after it is skipped in the restored multicast, you did not wait long enough.

Observation: The Parameters Are Not Critical In This Application.

EXTENSIONS

Operate Independent of the Source Protocol:

Real Audio Instead Of RTP --- DONE

1. Retransmit server encapsulates the source packets in RTP, and multicasts RTP.
2. Retransmit client reconstructs encapsulated multicast
3. Client removes encapsulation before performing remote multicast.

Cooperative Protocol With Multiple Retransmit Servers.

When all network servers are far from the source, recover packets from any server.

Integrate Recovery With Internet Bypass Circuits.

For Those 90% Loss Conditions.