Good Enough Header Compression (GEHCO)

Lucent Technologies

Tom Hiller

Pete McCann

Motivation

- 3GPP2 plans to support VOIP and multimedia directly to the mobile to provide new services
  - allows mobiles to use of SIP based browsers and land line VOIP/multimedia servers
- Size and frequency of RTP packets forces use of IP/UDP/RTP packet header compression

Categories of Header Compression

- **Transparent**: The IP/UDP/RTP header reconstructed by the decompressor matches the original IP/UDP/RTP sent packet bit for bit.
- **Non-Transparent**: The IP/UDP/RTP header reconstructed by the decompressor does *not* exactly match the original IP/UDP/RTP sent packet bit for bit. Some fields such as RTP sequence counts, time-stamps, and UDP CRC may be altered from an end to end point of view.

Transparent Compression

- Existing working group draft
- 3GPP2 example:1 byte of IP/UDP/RTP, 2 bytes for PPP, using variable rate vocoder (see draft for rates)
  - Reduced spectral capacity
    - 40% loss (synchronous to 20ms frames)
    - 16% loss (asynchronous to 20ms frames)
  - Another example: 1 byte and no PPP bytes: 13% loss
- Bottom line:
  - 3PGG2 will support transparent compression for those that need it and can pay for it
  - Unacceptable to 3GPP2 carriers for wide spread use

Header Stripping and Regeneration

- Send 20ms vocoder frames without headers
  - Maximizes spectral efficiency
- Methods
  - Gateway
  - Non transparent compression
    - Native wireless signaling
    - Link (PPP) level compression

Gateway Approach

- Mobile sets up a circuit path to gateway
  - The gateway is between the wireless link and the FA
  - Mobile and gateway agree on an over-the-air circuit connection
  - Gateway wraps vocoder data in IP/UDP/RTP header, using the gateway address as source address
- Problem
  - Does not work well with Mobile IP because IP/UDP/RTP packets have gateway address

Non Transparent Compression

- Need to send full header context information and associated circuit identifier from compressor to decompressor
  - Radio specific signaling
  - PPP

Data Link

- Extend PPP to
  - carry full header information with
  - identify an over-the-air connection that carries “headerless” vocoded data
- Decompressor wraps vocoder data in IP/UDP/RTP headers

Good Enough Header Compression

- General idea
  - Compress: Remove the IP/UDP/RTP header and transmit only the IP/UDP/RTP payload
  - Decompress: Wrap the received payload in a IP/UDP/RTP header
  - Use physical framing, not PPP/HDLC for RTP payload
- Result: Same spectral capacity as circuit voice
- IPR statement: There is no IPR

Link Assumptions

- The physical frame carries one vocoder frame, exactly
- The decompressor is able to determine the size of the vocoded data from the frame
- The frame rate is precisely known (e.g. 20ms)
- Mobile and network can share an identifier to can identify an over-the-air connection that will carry the vocoder data

Link Assumptions (contd)

- Two over-the-air connections
  - voice connection for vocoded voice
    - no retransmission for minimal latency
    - no IP or PPP headers
    - called the “vocoder connection” in the draft
  - data connection for TCP like data
    - retransmission for better effective error rate
    - carries IP and PPP headers
    - called the “PPP connection” in the draft

Operation

- Mobile establishes PPP, negotiates GEHCO, establishes vocoder over the air connection
- New RTP flow arrives at PPP layer entity
- The PPP entity stores src/dest address/ports and RTP sequence number in a table
- Compressor sends GEHCO_FULL_HEADER packet to decompressor
- Commences to send payload on over-the-air connection

Operation (continued)

- GEHCO_FULL_HEADER has
  - IP addresses
  - UDP ports
  - RTP sequence number
  - Connection ID (CID) = over-the-air connection ID
- Decompressor uses GEHCO_FULL_HEADER to create a new table entry
- Decompressor sends GEHCO_ACK to compressor

Decompression

- Extracts payload from over-the-air connection
- Creates time stamps and sequence numbers
  - time stamps created by counting physical frame intervals (e.g. 20ms intervals)
- Erred vocoder frames: Decompressor increments sequence counter
  - Allows RTP correspondent node to notice frames are missing and possibly mask the error/loss
  - Buffer data to prevent under-run due to IP jitter

Conclusion

- 3GPP2 will support IETF transparent compression, but needs a non transparent approach consistent with PPP
- GEHCO is a starting point
  - PPP provides for negotiation and context exchange
  - GEHCO defines PPP signaling for negotiating any of the other RTP compression formats
  - IPR statement: There is no IPR
- Can we make GEHCO a working group item?