Robust Header Compression (ROHC)

A step towards all-IP wireless networks

Carsten Bormann

TZI

ISSLOW: Integrated Services over slow links

Background: RTP is replacing TDM

ISSLOW: 1996 initiative for packet multimedia over serial

- 1) Low-speed links blocked by large frames:
  1500 Bytes == 400 ms (@ 28.8 kbit/s) ➜ Latency!
  - Provide fragmentation, suspend/resume: RFC 2686..2689
- 2) Header Overhead:
  44 Bytes @ 50 frames/sec == 17.6 kbit/s
  - A) Switch to fewer, larger packets ➜ Latency!
  - B) Provide good header compression: RFC 2507, 2508, 2509
- 3) Hard to reserve bandwidth with unknown header requirements
  - Obtain compressibility hints from application: RFC 3006

Header Compression: e2e vs. hop-by-hop

- RTP header is 12 bytes
  - SSRC is constant, SN and TS increase predictably
- Proposal: end-to-end header RTP header compression
  - Compress at source, decompress at destination
  - Issue: The biggest header is IP (20 bytes), and there is UDP (8 bytes)
  - Reordering makes it hard to compress very efficiently
- Header compression schemes operate hop-by-hop
  - Can use ordering on single link
  - Can compress IP header as well (20/40 bytes for IPv4/6)
  - Can compress between sources and destinations that don’t care
    - *Localized complexity!*

Existing Header Compression Standards

- TCP/IP header compression (VJ HC) RFC 1144
  – Compresses many IP/TCP header pairs to 4 bytes
- IP header compression (née: IPv6 HC) RFC 2507
  – Compresses successive headers identified by protocol field
  – Works on simplex links (no negotiation, no feedback)
  – Stops with UDP header (no further protocol field)
- Casner/Jacobson: CRTP RFC 2508
  – Can compress IP/UDP/RTP or just IP/UDP
  – Identify RTP by heuristics: more aggressive than IPHC
  – Requires duplex links (error feedback)
  – Still loss-less (e.g., preserves UDP checksum, if present)
- CRTP now “plugs” into IPHC RFC 2509
  – together with a PPP mapping document

Header Compression: Status end of 2000

- VJ HC has been available for a long time
- CRTP implementations now in the leading products
- PPP/IP/UDP/RTP now qualifies as an efficient method to run multimedia information over serial lines
  - no need for TDM style multiplexes any more
  - no problems with integration of data and multimedia
- Adopted for wireless
  - 3GPP references RFC2507 in R ’99

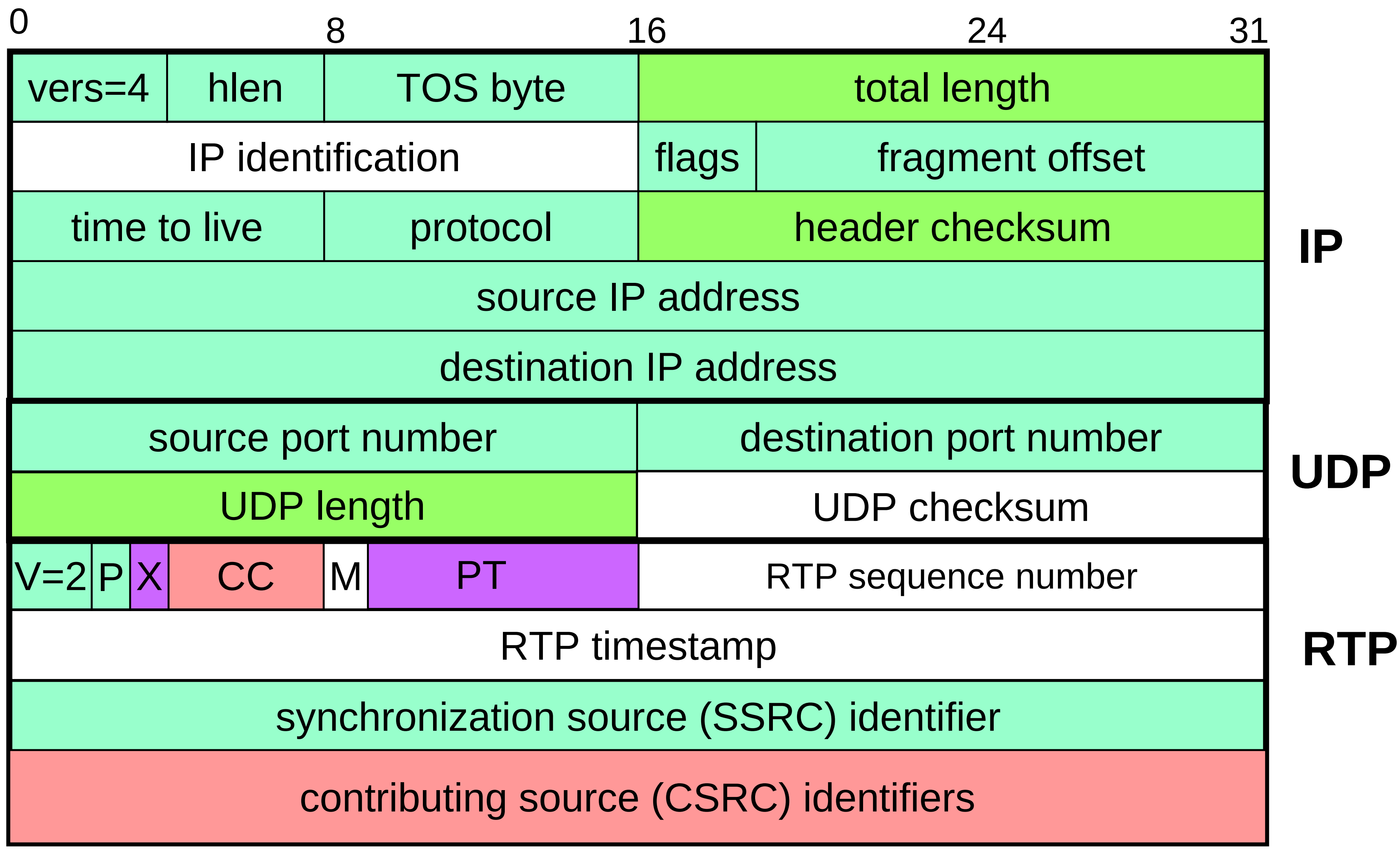
Enter Wireless: The need for ROHC

- **CRTP Issue: Robustness**
  - Delta coding works best on loss-free links
  - One loss ➨ inconsistency!
  - CRTP repair mechanism (CONTEXT-UPDATE) needs a round-trip
- **Loss propagation**
  - Losing one packet causes losing a round trip’s worth
  - Wireless: high error rate, large RTT
- **Damage propagation**
  - Not really an issue for PPP (16-bit or 32-bit CRC)
  - Higher spectrum efficiency calls for shorter checksums on 3G
  - Residual bit errors create long strings of bad packets
  - RFC2508 not appropriate for high-delay, high-loss links

IP + UDP + RTP header (20+8+12 byte)

<table>
<tr><th colspan="8">0</th><th colspan="8">8</th><th colspan="8">16</th><th colspan="7">24</th><th colspan="1">31</th><th></th></tr>
<tr><td colspan="4">vers=4</td><td colspan="4">hlen</td><td colspan="8">TOS byte</td><td colspan="16">total length</td><td rowspan="5">IP</td></tr>
<tr><td colspan="16">IP identification</td><td colspan="3">flags</td><td colspan="13">fragment offset</td></tr>
<tr><td colspan="8">time to live</td><td colspan="8">protocol</td><td colspan="16">header checksum</td></tr>
<tr><td colspan="32">source IP address</td></tr>
<tr><td colspan="32">destination IP address</td></tr>
<tr><td colspan="16">source port number</td><td colspan="16">destination port number</td><td rowspan="2">UDP</td></tr>
<tr><td colspan="16">UDP length</td><td colspan="16">UDP checksum</td></tr>
<tr><td colspan="2">V=2</td><td colspan="1">P</td><td colspan="1">X</td><td colspan="4">CC</td><td colspan="1">M</td><td colspan="7">PT</td><td colspan="16">RTP sequence number</td><td rowspan="4">RTP</td></tr>
<tr><td colspan="32">timestamp</td></tr>
<tr><td colspan="32">synchronization source (SSRC) identifier</td></tr>
<tr><td colspan="32">contributing source (CSRC) identifiers</td></tr>
</table>

no change + redundant fields

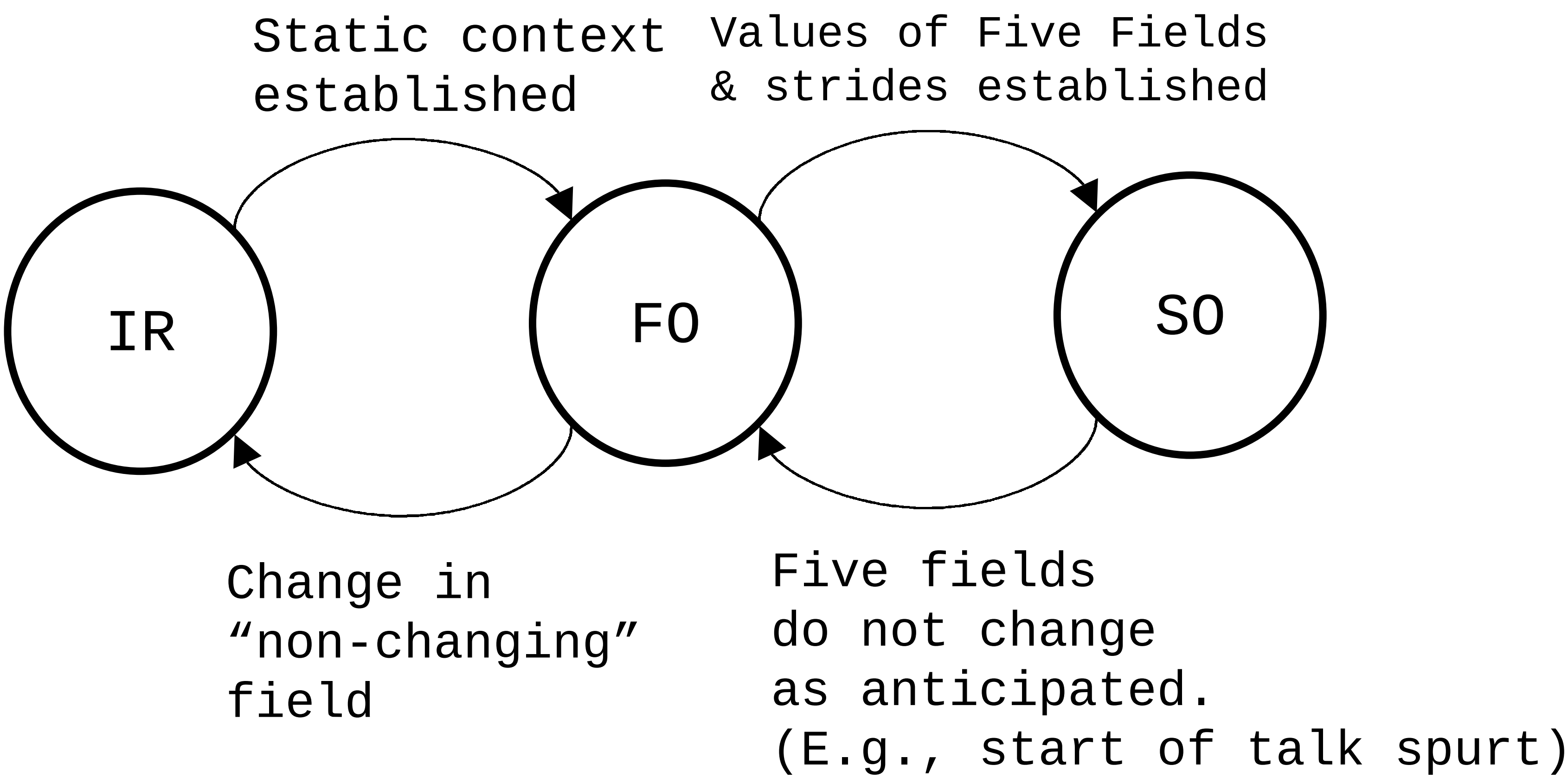


Basic Function

Two kinds of header fields:

- **Essentially constant:**
  - Context-identifying (IP addresses, ports, protocol…)
  - Rarely changing (TTL, TOS, Payload type)
- **Dynamic (“the Five Fields”):**
  - IP ID: Usually increments by 1 (or 256)
  - UDP checksum: Essentially random (or constantly zero)
  - RTP Marker bit: set once per talkspurt
  - SN (RTP sequence no): increments by 1
  - TS (RTP timestamp): increments by *TS_STRIDE*
    - (or more between talkspurts)

Compressor states



ROHC robustness (1)

- Do not use delta coding!
  - LSB coding (modulo) as a robust alternative
  - E.g., 4 to 6 bits are sufficient for the SN
  - Allow some variable-length coding for unusual cases
- Instead, use SN as "kernel field"
  - Send it with every packet
  - Losses or pre-compressor reorderings are apparent!
- Other fields: express as f(SN)
  - Characteristics of f are established in FO/SO state
  - E.g., for time stamp: f(SN) = TS_STRIDE*SN + TS_OFFSET (simplified)

ROHC robustness (2)

How to ensure state synchronization in the presence of losses and residual bit errors?

- **A) Saturation (as with RFC2507): "unidirectional mode"**
- **B) Optimism and Check: "optimistic mode"**
  - Send a CRC of *uncompressed* packet with each packet
  - Repeat changes often enough (limited saturation)
  - CRC catches the rest (1 RTT loss propagation!)
  - CRC also catches many residual bit errors
    - Ambiguity
- **C) Pessimism and Acknowledgements: "reliable mode"**
  - Do not assume state change at decompressor until acknowledged
    - 1 RTT of less efficient operation
  - Variant: Can play optimistic while waiting for ACK

ROHC framework

- Multiple *contexts* per *channel* (CID = context ID)
- Each context in use is bound to a *profile*
  - Set up by IR (initialization and refresh) packets
  - Currently defined: uncompressed, RTP, UDP, ESP
- Can define new profiles later
- Common packet types:
  - Short CID-extender
  - Feedback
  - IR/IR-DYN common prefix (must work on any context)
  - Segmentation protocol and padding

ROHC: The Result

- Can compress most headers to 1 byte
  - Unidirectional and optimistic mode:

```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| 0 |        SN         |  CRC      |
+===+===+===+===+===+===+===+===+
```

  - Reliable mode:

```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| 0   0 |            SN             |
+===+===+===+===+===+===+===+===+
```

- Robust against up to 12 (or >30) losses in sequence
  - Measurements indicates long loss trains are rare
  - Optimized for typical 3G style wireless voice (or video) links
    - (100 ms RTT, 20 ms frames, < 200 ms handover, ~ 1 s avg talkspurt)
  - Good transparency
- Support for IPv4/IPv6, most extension headers, IPSEC, GRE
- Draft –07 out this week ➨WG/IETF last call

Future

- **0-byte solutions:**
  - Use the tight radio frame timing to indicate SN/TS progress
  - Needs separate channel for non-SO packets
  - Gets rid of uninspired “header stripping” proposals
  - Requires buffering/resequencing at compressor
- **ROHC TCP:**
  - The requirements for robustness are maybe less stringent
    - Can do retransmission at link layer (see PILC)
  - Less stringent time constraints on development
  - New problems: Options like SACK, timestamps

All IP Wireless