Approaches to Difficult Aerospace Telecommunications Links

Joseph Ishac
jishac@nasa.gov
NASA Glenn Research Center
What qualifies as a “Difficult Link”

- **Straightforward issues**
  - Low Capacity
  - High Latency
  - High Losses
  - Low Reliability

- **More complex situations**
  - Multiple links between endpoints
  - Reordering
  - Asymmetric links
Formidable Links and Where to Find Them

- Yes, they do exist!
- Two frequent sources
  - As a result of physical environment
  - As a result of sharing spectrum
    - Highly contentious, sparse resource
    - Number of expected users
    - Licensed vs Unlicensed
- Examples
  - Satellite links, Deep Space Links
  - Command and Control Links
What makes these links so difficult for the network and users?

• Network protocols are designed to provide certain features/functions
  - TCP: Reliable, In-Order Data Stream (and other features)
• These protocols employ different methods to work
  - TCP: Connections, relies on feedback for control
• A protocol can suffer issues with certain link conditions
  - TCP: Does not do well with very high losses or high delay…
• Solutions introduce trade-offs, attempts to re-insert functionality
  - TCP not viable: Use UDP?
Network Layers

- TCP example shows impacts to the transport layer
- Switching from TCP to UDP impacts the application layer
  - Rewriting code
  - Need to pay attention to best practices – RFC 8085
- Protocols at all layers are impacted by these links

| Application | Transport | Internet | Data Link | Physical |

- Layer abstraction?
- How do we keep our links fully utilized and efficient?
Another Issue: Too many solutions

- Smart folks have thought up a lot of different solutions
  - Alphabet Soup of Solutions: DTN, MPTCP, DCCP, SCTP, ROHC, …
  - Solutions at every layer...

- Some users re-invent functionality
  - Check out my new custom Protocol Z over UDP for …

- So there are a lot of solutions – and that is a problem
  - Usually not a single solution that solves all the users problems
  - May not stack cleanly: Are at odds with each other in some way
  - Actually introduce other issues or complications
  - Almost always end up with security implications
## Issue #1: Low Capacity

<table>
<thead>
<tr>
<th>Media</th>
<th>Capacity</th>
<th>Time to process a single 1500 byte packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gigabit Ethernet</td>
<td>1,000,000,000 bits/s</td>
<td>0.000012 s</td>
</tr>
<tr>
<td>Average Cellular 4G LTE</td>
<td>30,000,000 bits/s</td>
<td>0.000400 s</td>
</tr>
<tr>
<td>Dial-Up 56K Modem</td>
<td>56,000 bits/s</td>
<td>0.214286 s</td>
</tr>
<tr>
<td>UAS Mode D Control</td>
<td>34,400 bits/s</td>
<td>0.348837 s</td>
</tr>
<tr>
<td>UAS Mode A Control</td>
<td>4,680 bits/s</td>
<td>2.564103 s</td>
</tr>
<tr>
<td>Iridium® Modem</td>
<td>2,400 bits/s</td>
<td>5.000000 s</td>
</tr>
</tbody>
</table>
Network Overhead vs. Capacity

- **Brief summary of Unmanned Systems in the National Airspace Goals**
  - Design future proof network [IPv6]
  - Ground operators can reach a mobile platform [Mobile IPv6]
  - Communications are secured [IPsec, ESP, AH]
  - Small unidirectional data messages [UDP]

- Network Headers alone total 117 bytes!
Low Capacity Potential Solutions

- **Header Compression [ROHC]**
  - Can significantly shrink headers. For example, 80 bytes to just 2
  - Impacted by how and when encryption is done
  - Risk of increased loss depending on what you compress

- **Shrink TCP Initial Congestion Window**
  - RFC 3390 (4K) and RFC 6928 (10 segments) are over sized
  - Troublesome if links have a large range of performance

- **Waiting for faster links is not a solution [Moore’s Law]**
  - Iridium® first available in November 1998 (20 years later!)
  - UAS CNPC links being designed now for 2020 and beyond.
### Issue #2: Latency

<table>
<thead>
<tr>
<th>Media</th>
<th>Approximate Distance</th>
<th>One Way Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Network</td>
<td>100 m</td>
<td>&lt; 0.001 s</td>
</tr>
<tr>
<td>Coast-to-Coast</td>
<td>4,000 km</td>
<td>0.030 s</td>
</tr>
<tr>
<td>Trans-Atlantic</td>
<td>4,600 km</td>
<td>0.035 s</td>
</tr>
<tr>
<td>Low Earth Orbit</td>
<td>781 km</td>
<td>0.003 s</td>
</tr>
<tr>
<td>Geosynchronous Orbit</td>
<td>35,786 km</td>
<td>0.120 s</td>
</tr>
<tr>
<td>Moon</td>
<td>384,402 km</td>
<td>1.282 s</td>
</tr>
<tr>
<td>Mars (Near) [2018-07-27]</td>
<td>57,774,698 km</td>
<td>(&gt; 3 min) 192.716 s</td>
</tr>
<tr>
<td>Mars (Far)</td>
<td>401,000,000 km</td>
<td>(&gt; 22 min) 1337.592 s</td>
</tr>
</tbody>
</table>
Latency Factors

- Iridium® One-Way Delay is about 750 ms or larger (not 3 ms)
  - Delay impacted by more than just range or line-of-sight
  - Processing, encoding, satellite-satellite routing, ground station location

![Diagram showing latency factors](image)
Latency Potential Solutions

● Can change the propagation speed to a point
  - Different mediums propagate faster… doesn't help for space

● Smarter queue management [SFQ, RED, AQM]
  - Gives feedback on congestion to all flows
  - Keep queuing delays to a minimum

● Shrink MTU
  - Increases responsiveness and fairness between flows
    - Oh yeah, it’s not just us out there.
  - Risk of fragmentation [PMTUD]

● Rate Limiting
Fair, Rate-Limited, Head Drop Queues

Which is better for TCP?

Tail Drop Queuing

4 + 3 2 1 = 3 2 1

Head Drop Queuing

4 + 3 2 1 = 4 3 2

One token added every N seconds

Queue added for each new connection

Max Fill

Segment released every N seconds

Initial Burst

Network Interface
Issue #3: High Losses

- Wireless data transmission is prone to errors
  - Bit flips, bit insertion, bit loss…
  - Quality can vary depending on conditions (ie: weather) and distance
  - Erred data packets are usually discarded on reception

- Mistaking delays for loss
  - Iridium® one way minimum delay of roughly 750 ms (1.5 s round trip)
  - RFC 6298 (TCP retransmit timer) reduces initial timeout to 1 seconds
  - Unnecessary retransmissions are costly (due to low capacity)
  - Protocols such as TCP suffer, mistaking loss for congestion
High Losses Potential Solutions

- Use of Forward Error Correction (FEC) [Reed-Solomon, LDPC]
  - Usually applied to the link – Corrects errors “in-flight”
  - Stronger encoding/decoding usually takes longer
  - Adds latency

- Accurate or longer timers
  - Hard problem to balance responsiveness and accuracy
  - Sometimes changing system defaults are hard
    - RFC 6298 timeout of 1 second is hard coded into the Linux kernel
      `#define TCP_TIMEOUT_INIT ((unsigned)(1*HZ))`
Issue #4: Low Reliability

- In addition to losses, links may fail completely
  - Out of range
  - Loss of coverage
  - Lack of resources
- Transitions may be frequent
- What does that mean for:
  - Protocols
  - Connections
  - State
Iridium® Link Transitions

- **Flight Date:** November 18, 2016
- **Flight Duration:** 13 Hours
- **Events that changed the number of active links:**
  - 325 (25 changes / hour)
- **Nearly one fourth of the flight is in a degraded state**

<table>
<thead>
<tr>
<th>Number of Active Links</th>
<th>Seconds</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>35643</td>
<td>76.26%</td>
</tr>
<tr>
<td>3</td>
<td>8269</td>
<td>17.69%</td>
</tr>
<tr>
<td>2</td>
<td>1969</td>
<td>4.21%</td>
</tr>
<tr>
<td>1</td>
<td>235</td>
<td>0.50%</td>
</tr>
<tr>
<td>0</td>
<td>624</td>
<td>1.34%</td>
</tr>
</tbody>
</table>
Low Reliability Potential Solutions

- Adaptive modulation to increase range
  - Capacity decreases
- Use multiple or alternate connections
- Typical Iridium® usage will deploy 4 or more modems
  - Currently 4 channels are used to provide a total of 9.6 Kbit/s
  - Decreases probability of having no links available
    - In the flight example, only 1.34% of the flight had no connection
- Unfortunately multiple links create their own set of issues
Issue #5: Multiple links between endpoints

- N-times the same type of link or
  Multiple types of links
  - Often used to increase capacity or used as a fail over
  - Appears at N interfaces
  - Example: Wi-Fi + Cell
  - Example: Iridium®
    - Image shows 4 modems
- Hard managing data over multiple streams
Multiple Links Potential Solutions

- **Link Level Bonding**
  - Appears as one physical interface
  - Hides issues from other layers
    - TCP reacting to losses isolated on only a single bad link

- **Bonding at Upper Layers**
  - Identify and isolate link issues
  - More complex state
  - Solution not as universal
    - MPTCP good for TCP only

<table>
<thead>
<tr>
<th>Application</th>
<th>Web Requests, Internet Relay Chat, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport</td>
<td>TCP</td>
</tr>
<tr>
<td>Network</td>
<td>IP</td>
</tr>
<tr>
<td>Data Link</td>
<td>MLPPP</td>
</tr>
<tr>
<td></td>
<td>PPP          PPP    PPP    PPP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Web Requests, Internet Relay Chat, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPTCP</td>
</tr>
<tr>
<td>TCP</td>
</tr>
<tr>
<td>IP</td>
</tr>
<tr>
<td>PPP</td>
</tr>
</tbody>
</table>
Issue #7: Asymmetric links

- Links are not always the same capacity in each direction
  - Similar to how home internet has different upload and download speeds
- Satellite links can have large down-link to up-link ratios
  - Further exaggerated by long delays
- Really impacts protocols such as TCP
  - May limit the TCP ACK stream and overall rate
  - Feedback constrained on return channel
- Possible Solutions
  - Different types of feedback algorithms or mechanics [NACKs]
  - Unidirectional communication
Issue #6: Reordering

- Reordering does happen
  - Multiple links (Dumping a queue on a failed link)
  - Multiple paths
  - Changing delays
  - Even a single link is not immune

- Possible Solutions
  - Use protocols that have a mechanism to deal with reordering (like TCP)
  - Handle ordering in the application
  - Is order even important? May not be to some applications
  - Request: Please do not assume in-order delivery (even in slides)
Thank you

Questions?
Backup Slides
Glossary of Terms and References (In order of appearance)

- Iridium® - https://www.iridium.com/
- GPS – Global Positioning System
- ISS – International Space Station
- UAS – Unmanned Aircraft Systems
- ATC – Air Traffic Control
- TCP – Transmission Control Protocol
- UDP – User Datagram Protocol
- RFC 8085 - UDP Usage Guidelines
- DTN – Delay/Disruption Tolerant Networking
- MPTCP – MultiPath Transmission Control Protocol
- DCCP – Datagram Congestion Control Protocol
- SCTP – Stream Control Transmission Protocol
- ROHC – Robust Header Compression
- IPv6 – Internet Protocol Version 6
- IPsec – Internet Protocol security
- ESP – Encapsulating Security Payload
- AH – Authentication Header
- RFC 3390 – Increasing TCP's Initial Window
- RFC 6928 – Increasing TCP's Initial Window
- SFQ – Stochastic Fairness Queuing
- RED – Random early detection
- AQM – Active Queue Management
- MTU – Maximum Transmission Unit
- PMTUD – Path MTU Discovery
- RFC 6298 – Computing TCP’s Retransmission Timer
- FEC – Forward Error Correction
- Reed-Solomon
- LDPC – Low Density Parity Checks
- NACK – Negative Acknowledgments