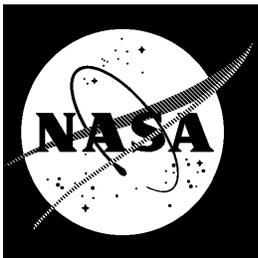


**Performance Efficiency of Internet
Protocol (IP) in Space
Applications**

Revision 0

November 16, 1999

Produced By
Edward Criscuolo,
Computer Sciences Corp.
for the
Goddard Space Flight Center
Under SOW #96964



National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, Maryland

Section 1 . Introduction

1.1 Purpose

Space based systems have constraints on power and weight that translate into constraints on available communications bandwidth. This document addresses the performance efficiency of the Internet Protocol (IP) suite when used as the direct communication protocol for space missions. It specifically addresses the proposed use of User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) at the transport layer, IP at the network layer, and High-level Data Link Control (HDLC) at the link layer. Comparisons are made with equivalent communication services as defined by the Consultative Committee for Space Data Systems (CCSDS) and currently in use.

1.2 Scope

This document *only* addresses the raw bit efficiency of the communication protocol. Reed-Solomon, Convolutional, and other coding techniques can be applied equally well to either IP or CCSDS protocols, so their overhead is not included. Latency and bandwidth-delay issues are not discussed here, but will be the topic of a separate report.

Section 2. References

2.1 Applicable Documents

ISO 3309, High-level Data Link Control (HDLC) - Frame Structure

ISO 4335, High-level Data Link Control (HDLC) - Consolidation of Elements of Procedures

CCSDS 101.0-B-4, Telemetry Channel Coding, Blue Book. Issue 4. May 1999.

CCSDS 102.0-B-4, Packet Telemetry, Blue Book. Issue 4. November 1995.

RFC-2507, IP Header Compression, February 1999

Section 3. Analysis

3.1 Background

Modern networking is based on the OSI Seven Layer Model of network architecture, a framework for international standards in heterogeneous computer network architecture. The OSI architecture is split between seven layers, from lowest to highest:

- 1 physical layer
- 2 data link layer
- 3 network layer
- 4 transport layer
- 5 session layer
- 6 presentation layer
- 7 application layer

Each layer uses the layer immediately below it and provides a service to the layer above. This analysis will be concerned primarily with the transport, network, and data-link layers.

3.1.1 Transport Layer

The transport layer determines how to use the network layer to provide a virtual, point to point connection so that host A can send messages to host B. The two transport protocols that will be examined are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). TCP is connection-oriented and stream-oriented, as opposed to UDP which is a connectionless protocol. The CCSDS packet telemetry concept that most closely matches this is the CCSDS Virtual Channel which is implemented in the CCSDS transfer frame.

3.1.2 Network Layer

The network layer determines routing of packets of data from sender to receiver via the data link layer and is used by the transport layer. The most common network layer protocol is IP, providing 32-bits of source address and 32-bits of destination address. CCSDS implements a single address in the 10-bit spacecraft Id field in the transfer frame header.

3.1.3 Data Link Layer

The data link layer splits data into frames, fragmenting as necessary, for sending on the physical layer. The protocol used in data link layer has a characteristic Maximum Transmission Unit (MTU) which determines whether a network layer packet must be fragmented. Typical MTUs are on the order of 1000 to 2000 bytes. Ethernet, for example, has an MTU of 1500. One of the most important data link control protocols is HDLC. Not only is HDLC widely used, but it is the basis for many other important data link control protocols, which use the same or similar formats and the same mechanisms as employed in HDLC. CCSDS protocols implement the data link layer with the CCSDS transfer frame.

3.2 Current Practice Using CCSDS Protocols

Under CCSDS protocols, telemetry data is organized into different kinds of packets, each kind identified by an ApID in the packet header. Various packet options exist, some with an optional secondary header, but all have a minimum of a 6 byte header. In addition to the ApID and length, this header contains some packet sequence information which maps into the transport layer. Multiple packets are collected into a transfer frame. Packets do not align synchronously within the transfer frame. See figure 3-1. Transfer frames are usually a fixed size which is driven by the Reed-Solomon coding used. For the (255,223) Reed Solomon code specified in CCSDS 101.0-B-4: “Telemetry Channel Coding”, this length is an integer multiple of 223 bytes, with a maximum size of 1115 bytes. A transfer frame has mandatory 8 byte header which combines the functions of the transport, network, and link layers. Some transfer frames contain an optional secondary header which we will ignore for purposes of this analysis. Finally, the transfer frame is synchronously packed into a Reed-Solomon coding block with a single 4 byte sync pattern and a 32 byte trailer of Reed-Solomon symbols per every 223 bytes of data. Since the Reed-Solomon (and any other coding) used will be identical for either CCSDS or IP, its overhead in each case will be the same (approx. 14%) and can be ignored. Thus, the overhead, excluding R/S, can be calculated for the best case as follows:

n = number of data bytes per packet

t = max. total bytes per transfer frame = 1115

p = number of packets per transfer frame = $(t - 8) / (n + 6)$

o = overhead bytes per transfer frame = $8 + 6p$

$$\begin{aligned} \text{percent overhead} &= 100 \cdot (o / t) \\ &= 100 \cdot [8 + 6((t - 8) / (n + 6))] / t \\ &= 100 \cdot [0.0071748 + (5.9569506 / (n + 6))] \end{aligned}$$

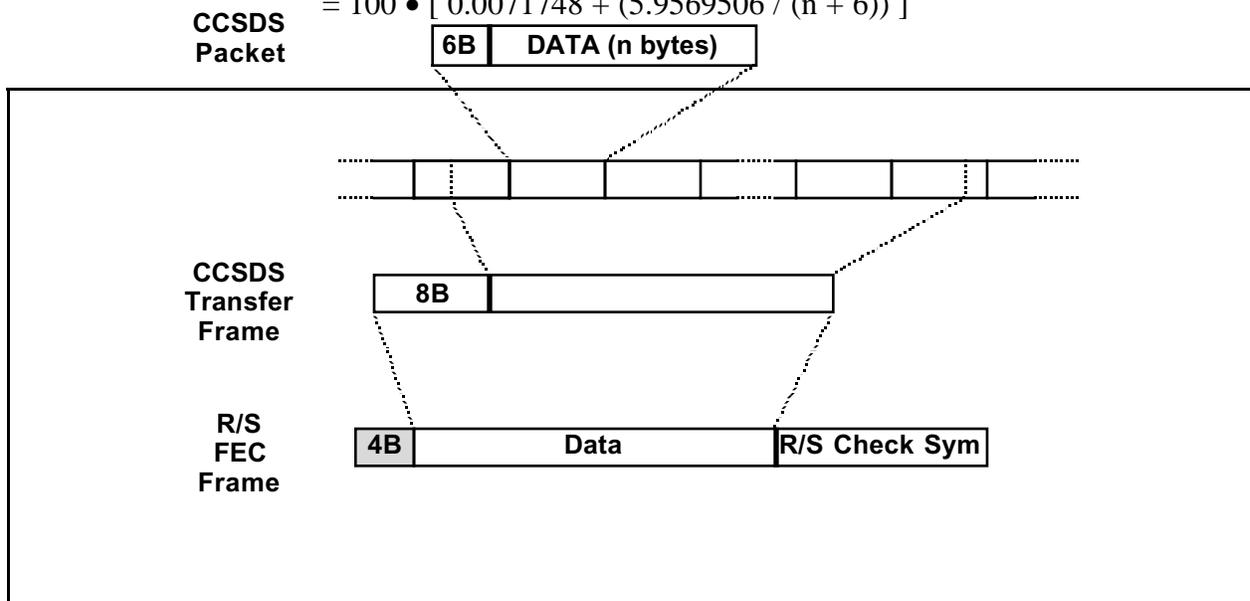


Figure 3-1.

3.3 Proposed Usage of TCP/IP/HDLC

Under TCP/IP, data at the application layer is sent as a reliable stream of bytes. For equality of comparison with CCSDS protocols, it will be assumed that telemetry data is organized into n-byte packets with an equivalent, if not identical, 6-byte header. See figure 3-2. At the transport layer, one or more whole application packets are synchronously collected into a TCP packet with a 20-byte header. For purposes of this analysis it will be assumed that there is one application packet per TCP packet. This is a worst-case choice for overhead, but was chosen to compare against the best-case analysis done for CCSDS in section 3.2. At the network layer, each TCP packet is put into an IP packet with a 20-byte header. At the link layer, each IP packet is put into an HDLC frame with a 1-byte sync, a 2-byte header, and a 2-byte trailer. Finally, the HDLC frames are aggregated asynchronously into 1115-byte Reed-Solomon coding block with a single 4 byte sync pattern and a 32 byte trailer of Reed-Solomon symbols per every 223 bytes of data. This is identical to the coding used in the analysis of CCSDS in section 3.2. Thus, the overhead, excluding R/S, can be calculated for the worst case as follows:

n = number of data bytes per packet

o = overhead bytes per HDLC frame = $5 + 20 + 20 + 6 = 51$

percent overhead = $100 \cdot [o / (o + n)]$
 = $100 \cdot [51 / (51 + n)]$

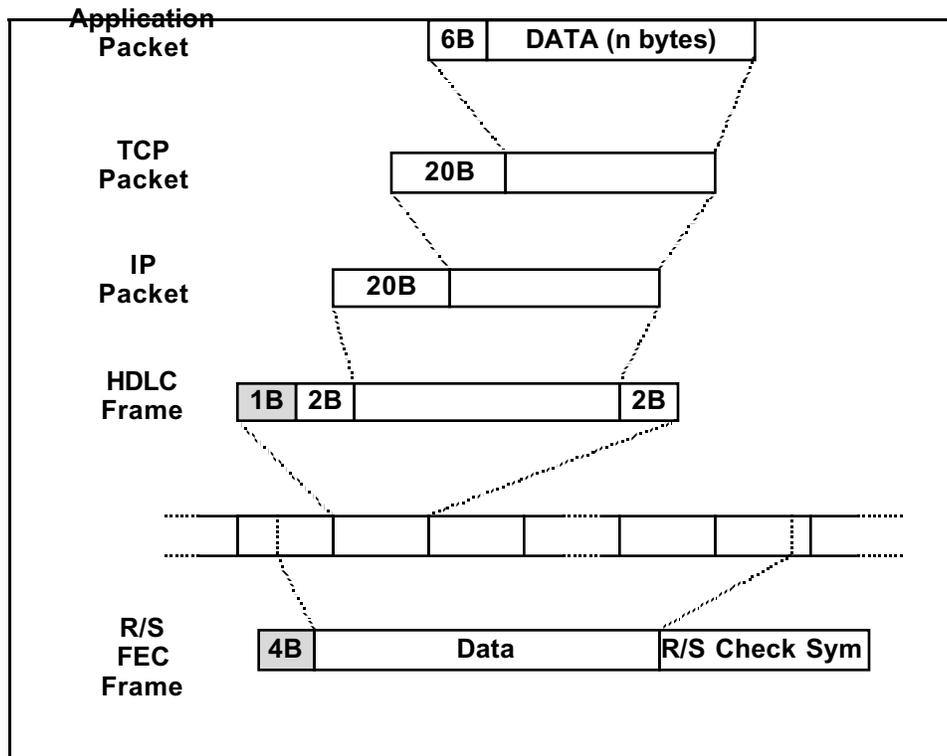


Figure 3-2.

3.4 Proposed Usage of UDP/IP/HDLC

Under UDP/IP, data at the application layer is sent as an atomic datagram without guaranteed order or delivery. This is functionally equivalent to CCSDS packets. For equality of comparison with CCSDS protocols, it will be assumed that each datagram of telemetry data is organized as an n-byte packet with an equivalent, if not identical, 6-byte header. See figure 3-3. At the transport layer, each application packet is placed into a UDP packet with an 8-byte header. At the network layer, each TCP packet is put into an IP packet with a 20-byte header. At the link layer, each IP packet is put into an HDLC frame with a 1-byte sync, a 2-byte header, and a 2-byte trailer. Finally, the HDLC frames are aggregated asynchronously into 1115-byte Reed-Solomon coding block with a single 4 byte sync pattern and a 32 byte trailer of Reed-Solomon symbols per every 223 bytes of data. This is identical to the coding used in the analysis of CCSDS in section 3.2. Thus, the overhead, excluding R/S, can be calculated as follows:

n = number of data bytes per packet

o = overhead bytes per HDLC frame = $5 + 20 + 8 + 6 = 39$

percent overhead = $100 \cdot [o / (o + n)]$
 $= 100 \cdot [39 / (39 + n)]$

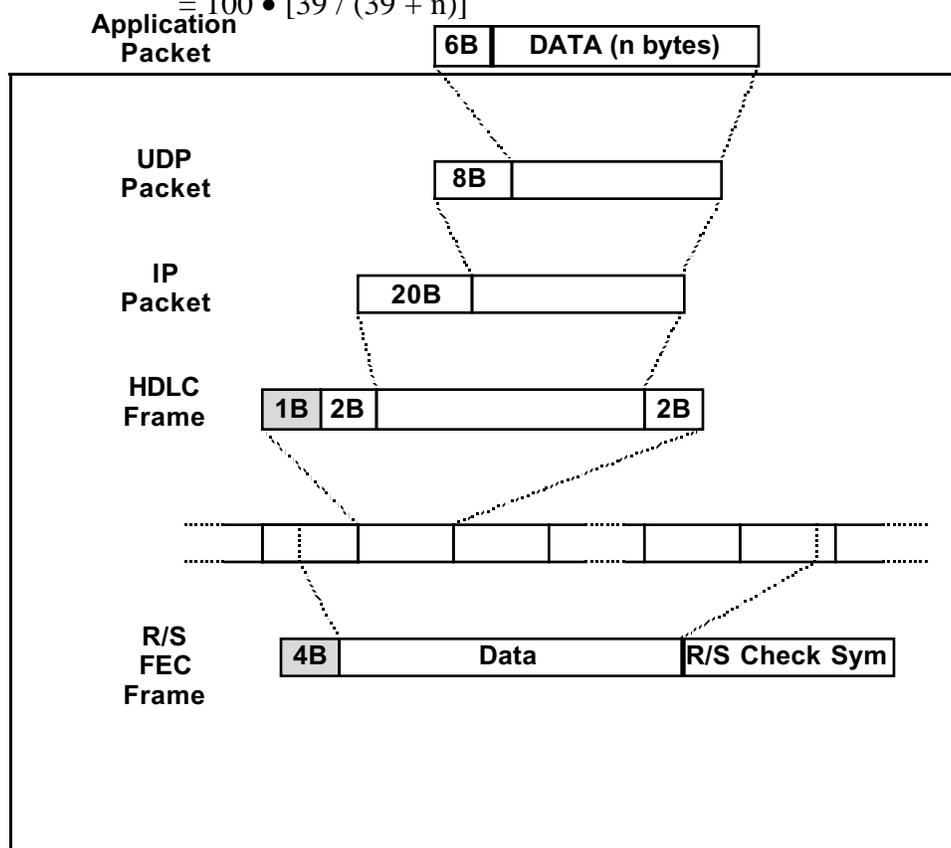


Figure 3-3.

3.5 IP Header Compression

The Internet Engineering Task Force (IETF) is in the process of completing a standard for header compression known as RFC-2507 "IP Header Compression". Headers of typical UDP or TCP packets can be compressed down to 4-7 bytes including the 2 byte UDP or TCP checksum. This largely removes the negative impact of large IP headers and allows efficient use of bandwidth on low and medium speed links. The compression algorithms are specifically designed to work well over links with nontrivial packet-loss rates. Several wireless and modem technologies result in such links.

A typical application of this standard takes the headers for the transport and network layers for either TCP/IP (40 bytes) or UDP/IP (28 bytes) and compresses approx. 90% of them down to 6 bytes.

Thus, the overhead for TCP/IP with header compression is:

$$\text{percent overhead} = 100 \bullet [(0.1 \bullet 51 + 0.9 \bullet 17) / (0.1 \bullet 51 + 0.9 \bullet 17 + n)]$$

and the overhead for UDP/IP with header compression is:

$$\text{percent overhead} = 100 \bullet [(0.1 \bullet 39 + 0.9 \bullet 17) / (0.1 \bullet 39 + 0.9 \bullet 17 + n)]$$

Section 4. Summary

4.1 Comparisons

Figure 4-1 shows the overhead comparison between small (100 byte) and average (1000 byte) packets for all of the protocols analyzed in section 3. Figure 4-2 shows a graph of all the results.

Protocol	100 Byte Packets	1000 Byte Packets
TCP/IP/HDLC	33.8%	4.9%
UDP/IP/HDLC	23.1%	3.6%
TCP/IP/HDLC w/hdr comp.	16.9%	2.0%
UDP/IP/HDLC w/hdr comp.	16.1%	1.9%
CCSDS	6.3%	1.3%

Figure 4-1.

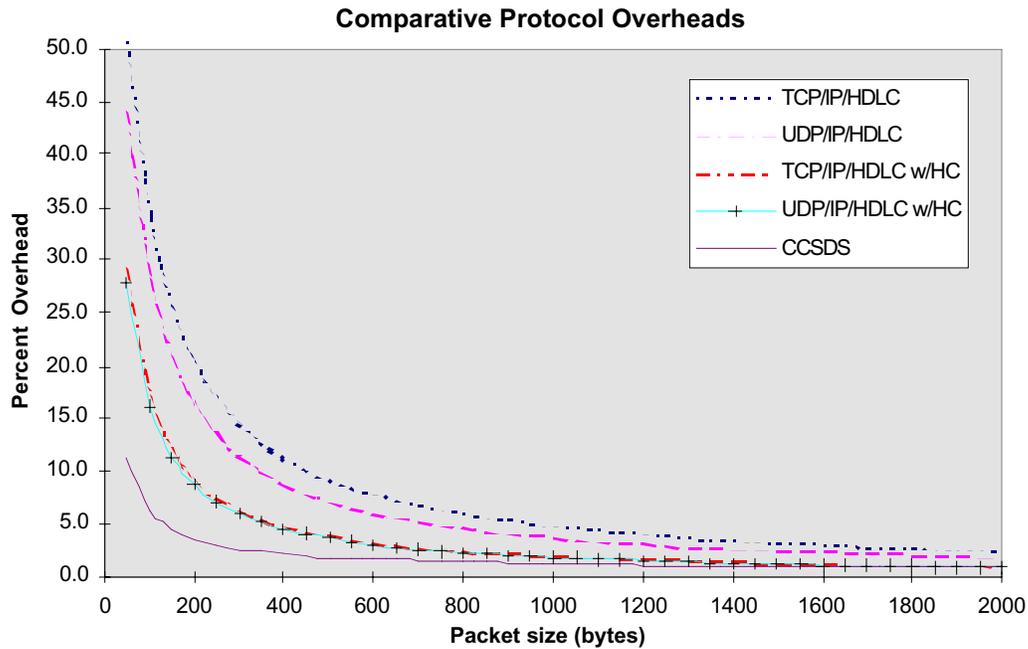


Figure 4-2.

4.2 Conclusions

With its real-time nature and non-guaranteed delivery, CCSDS packet telemetry most closely resembles UDP/IP. For nominal 1000 byte packets, UDP/IP's overhead is 3.6% as compared to CCSDS's 1.3%, a difference of only 2.3%. If UDP/IP with header compression is considered, the difference drops to only 0.6%. These minor differences are insignificant when compared to the Reed-Solomon overhead of 14%.

TCP/IP has a somewhat higher overhead of 4.9%, which is a difference of 3.6% above CCSDS; however, TCP/IP provides additional functionality over CCSDS packet telemetry in the form of automatic retransmission of lost or damaged packets. This will buy additional contact time by allowing operation closer to the horizon where contact becomes intermittent.