NETWORK TELEMETRY LINK THROUGHPUT MAXIMIZATION APPROACHES

Myron L. Moodie, Todd A. Newton, Ben A. Abbott
Southwest Research Institute®
San Antonio, Texas
myron.moodie@swri.org, todd.newton@swri.org, ben.abbott@swri.org

ABSTRACT

The use of Ethernet and Internet Protocol (IP) networking technologies in flight test instrumentation and telemetry systems is rapidly increasing, driven by the ubiquity, scalability, and flexibility of networking technologies. Networks first made a positive impact in ground station infrastructure and have recently been emerging in test article data acquisition infrastructure in programs such as the A380, 787, P-8A, and Future Combat Systems. The next logical step is to provide a two-way network telemetry link to fully extend the flexibility of the network between the test articles and ground station. The United States Department of Defense (DoD) integrated Network-Enhanced Telemetry (iNET) program is currently working to build a standardized network telemetry link for exactly this purpose.

When developing a network telemetry link, the limited availability of telemetry spectrum must be considered and thus it is critical to choose system-level approaches to maximize the throughput achieved from the link. This paper first presents the statistics of the network data that would typically use this link based on empirical data from current network-based flight test instrumentation systems. Several approaches to using a network telemetry link are then presented. Predicted achievable throughputs of each approach are presented that are derived from the statistics of the empirical test data. Based on this, the paper presents recommendations for building systems using network telemetry links.

KEYWORDS

iNET, network telemetry link, IP, throughput

INTRODUCTION

For over forty years, the IRIG 106 Chapter 4 standard for PCM-based test and telemetry systems has provided a consistent, standardized approach to facilitate flight test and other test and evaluation (T&E) operations. While this standard has served the T&E community well, it is

beginning to show serious limitations as the complexity of test articles and the amount of data collected from them continues to increase dramatically. These factors, combined with the growing ubiquity of network technologies driven by the rapid growth of the Internet and associated enterprise and home networks, have led to increased interest in the flexibility of network technologies within the T&E environment.

While PCM has worked well up to this point, it suffers from a fairly rigid physical and logical structure that does not allow the flexibility and scalability that a general network-based approach provides. This leads to fixed-function equipment, less-scalable measurement addition, and equipment interconnection and cabling complexity. Traditional PCM-based telemetry links provide only one-way (test article to ground) communications and thus do not allow dynamic reconfiguration of what data is being telemetered.

The use of Ethernet and Internet Protocol (IP) networking technologies in flight test instrumentation and telemetry systems is rapidly increasing, driven by the ubiquity, scalability, and flexibility of networking technologies. Networks first made a positive impact in ground station infrastructure and have recently been emerging in test article data acquisition infrastructure in programs such as the A380, 787, P-8A, and Future Combat Systems. The next logical step is to provide a two-way network telemetry link to fully extend the flexibility of the network between the test articles and ground station. The United States Department of Defense (DoD) integrated Network-Enhanced Telemetry (iNET) program is currently working to build a standardized network telemetry link for exactly this purpose.

The iNET Project was set up by the DoD Central Test and Evaluation Investment Program (CTEIP) to chart a forward path for telemetry. Some of the goals include the development of standards that provide device interoperability while supporting many different and potentially dynamic setups. Since the scope of the standards may include multiple vehicles, multiple types of vehicles or test articles, and multiple ranges, there is a very wide field of standardization to cover.

One of the key emphases of the iNET Project is to maximize the efficient use of limited telemetry spectrum through the addition of a two-way network telemetry link. Unlike traditional one-way PCM telemetry systems, a network telemetry link provides all of the flexibility of extending the logical network from the ground to the test article resources. Consequently, the data sent down the network telemetry link can be dynamically changed during the operation of a test. This two-way link opens up additional possibilities for control, configuration, and status of the instrumentation on the test article.

Network telemetry links provide a general purpose network connection between the ground and test article, but the capacity of the link is still typically small when compared to the amount of data available on the test article. Treating this link as “just a network” without regard for its capacity risks making inefficient use of this limited resource.

NETWORK TRAFFIC STATISTICS

The first step to evaluating network telemetry link performance is to understand the statistics of the types of traffic that are likely to transit the link in a telemetry system. Three common use cases for an iNET network telemetry link will likely be: transport of live acquired data, retrieval of recorded acquired data, and management (control, status, and configuration) of test article resources.

The first case will likely use the Latency/Throughput Critical (LTC) Delivery protocol defined in the iNET Test Article Standard. This protocol uses multicast User Datagram Protocol (UDP)/IP to transport acquired data in Telemetry Network Systems (TmNS) Data Messages. The second case will likely be a mix of standard File Transfer Protocol (FTP) and Reliability Critical (RC) Delivery protocol that is defined in the iNET Test Article Standard. The management data will use Simple Network Management Protocol (SNMP) as specified in the iNET System Management Standard.

Recent experience with network-based instrumentation systems with transport mechanisms similar to LTC Delivery protocol provides a basis for predicting the network traffic statistics of live acquired data. Figure 1 shows the distribution of UDP datagram sizes for a typical composition of acquisition sources.

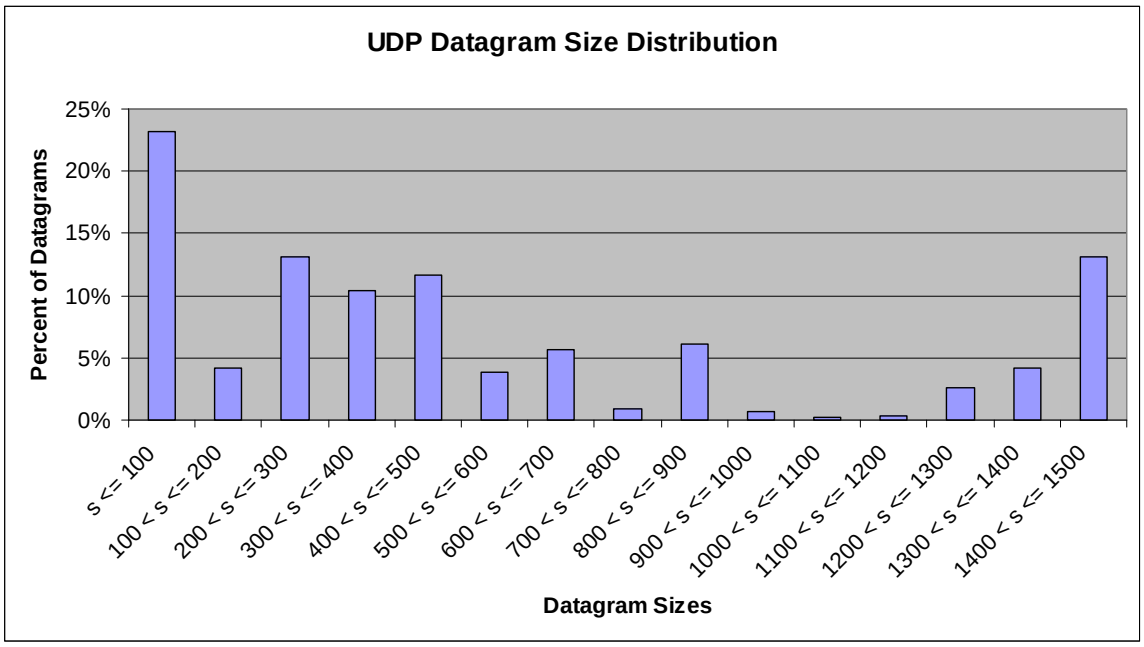

Figure 1. Acquisition Data Size Distribution

RC Delivery protocol uses a TCP/IP data channel combined with a Real-Time Streaming Protocol (RTSP) control channel. The network traffic shape of this traffic is expected to be similar to FTP which also uses separate TCP/IP data and control channels. Figure 2 shows the distribution of TCP segment sizes for an FTP transfer of a multi-megabyte file.

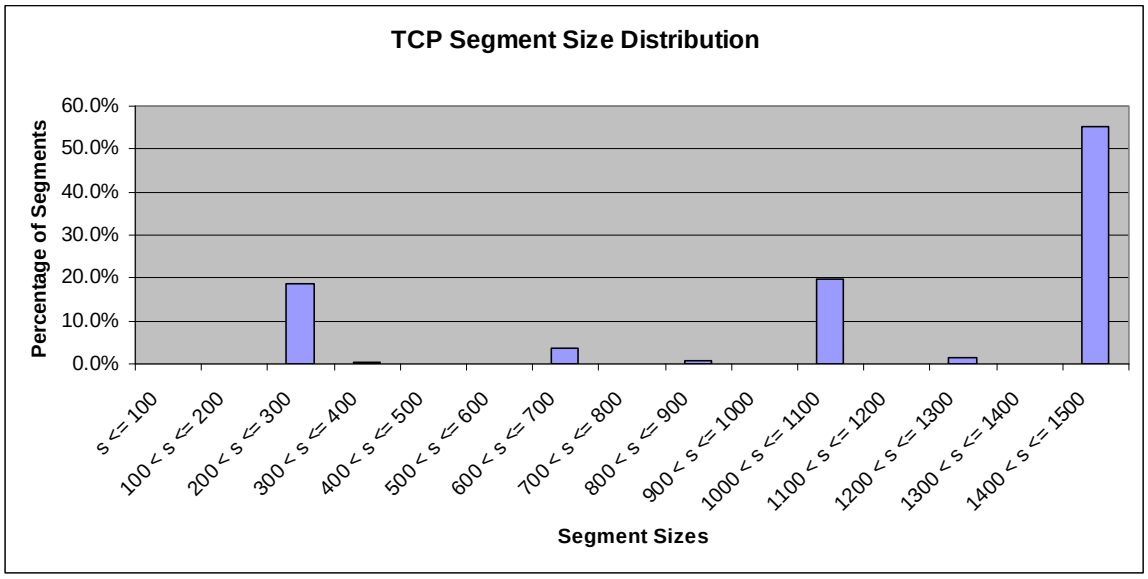

Figure 2. FTP Data Size Distribution

SNMP uses UDP/IP in a query/response and command/response application-layer protocol. Due to the relative efficiency of SNMP Object Identifier (OID) encoding (compared to textual OID names), the majority of these packets are small. Figure 3 shows the distribution of UDP datagram sizes for SNMP interactions with a typical composition of network-based instrumentation equipment.

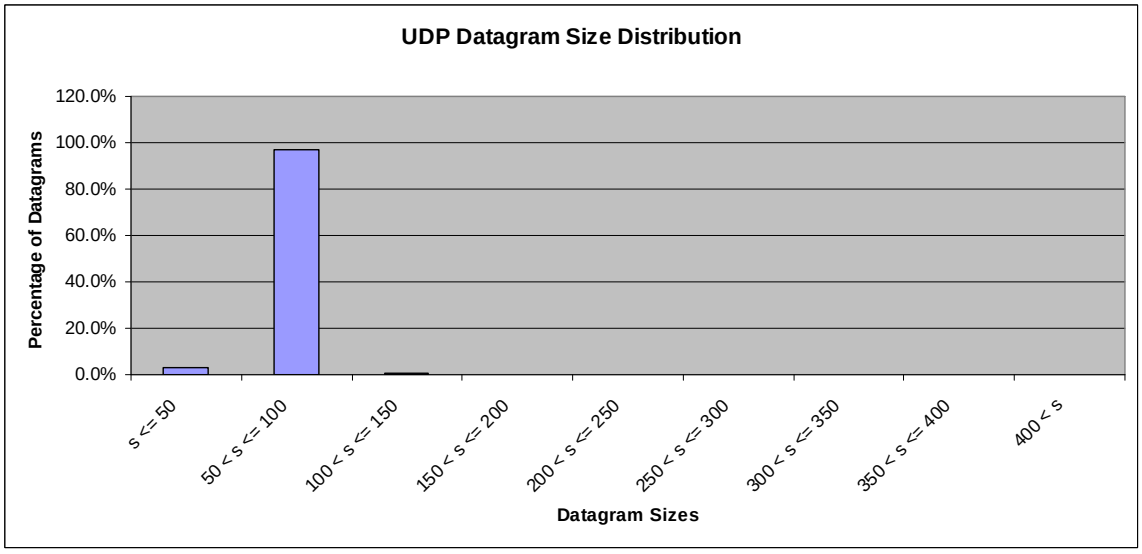

Figure 3. SNMP Data Size Distribution

NETWORK TELEMETRY LINK APPROACHES

Network telemetry links may have the same purpose and may even have the same external interfaces, but there are a number of approaches that can be used to implement the air interface. The obvious design options are related to modulation, channel access, channel coding, and error control coding. All of these parameters affect the overall performance of the network telemetry link, including maximum achievable throughput. However, there is another often overlooked design choice that can have a significant impact on maximum achievable throughput. The selection of what data is sent as the payload of the network telemetry link and the format of that payload is important to the overall usability of the network telemetry link.

This paper considers several approaches including Ethernet bridging, IP forwarding, point-to-point protocol, and application-level proxies. The first three approaches are mutually exclusive and determine the way the user data is encapsulated transiting through the network telemetry link. The fourth approach can be combined with any of the other three and allows downselecting the data transiting through the network telemetry link. Each of these approaches has advantages and disadvantages in the areas of implementation flexibility, overhead, and complexity. All of these approaches assume that the network telemetry link is used to connect two separate Ethernet/IP wired networks through the wireless network telemetry link. These two networks are referred to as the "near-end" and "far-end" networks relative to the transit of a particular network packet.

Ethernet Bridging

Ethernet bridging approaches use the entire Ethernet frame that arrives at the near-end network as the payload data. After passing through the network telemetry link, the telemetry transceiver regenerates the exact same Ethernet frame onto the far-end network. This connects the near- and far-end networks at the data link layer (layer 2) in the same way as if the networks were connected with a wired bridge or layer 2 switch or with an IEEE 802.11 bridge.

The advantage to this approach is that the telemetry transceivers do not need to process any portion of the Ethernet frames outside of the Ethernet headers, which reduces transceiver complexity. This also means that the network telemetry link can carry any Ethernet network data, so non-IP protocols like AppleTalk, NetBIOS, and Netware Internetwork Packet Exchange (IPX) can be transported alongside IP. Virtually all modern applications only need to support IP, so this capability is generally not needed.

Carrying the whole Ethernet frame, however, also means that a portion of the network telemetry link is consumed carrying Ethernet header/trailer fields that could be generated locally by the wired side of the telemetry transceivers if a higher-layer approach is used. The Ethernet headers/trailers add 18 bytes per Ethernet frame transported, as shown in Figure 4. For minimum size 64-byte Ethernet frames, these headers/trailers account for 28.1% of the user data being transported. For maximum size 1518-byte Ethernet frames, this is a more tolerable 1.1%.

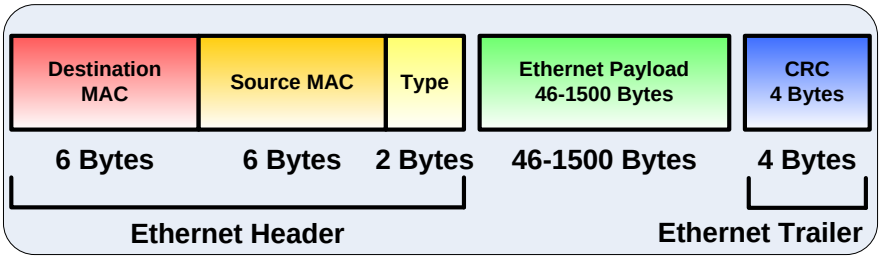

Figure 4. Ethernet Frame Structure

Another disadvantage is that this approach will transport all Ethernet frames arriving at the near-end transceiver, whether they are needed on the far-end network or not. This includes broadcast protocols such as Spanning Tree Protocol (STP), address resolution protocol requests for near-end nodes, and unsubscribed multicast traffic that is not filtered by an Internet Group Management Protocol (IGMP)-snooping capable switch connected to the telemetry transceiver. This extra traffic can consume a significant portion of the network telemetry link depending on the capacity of the link.

IP Forwarding

Instead of transporting Ethernet frames, the network telemetry link can be built to only transport IP packets. In this case, the near-end transceiver would remove the Ethernet header/trailers and transport the IP packet to the far-end transceiver. The far-end transceiver would generate a new Ethernet frame(s) to the far-end network with the IP packet as a payload. Basically, the network telemetry link would route IP packets instead of bridging Ethernet frames. The near-end transceiver would serve as an IP gateway for the near-end network and the far-end transceiver would serve as an IP gateway for the far-end network.

This approach has the advantage of reduced overhead compared to the Ethernet bridging case at the cost of the telemetry transceivers needing to process IP (layer 3). This is likely not a significant cost since this basic routing can be implemented in a small, embeddable processor. This approach would carry the overhead of the IP headers, which consume 20 to 60 bytes (depending on if options are used) per IP packet, as shown in Figure 5. Since an IP packet can legally carry a zero-byte payload, the IP headers account for between 0.03% (for a maximum length 65535-byte IP packet) and 100% of the IP packet traffic.

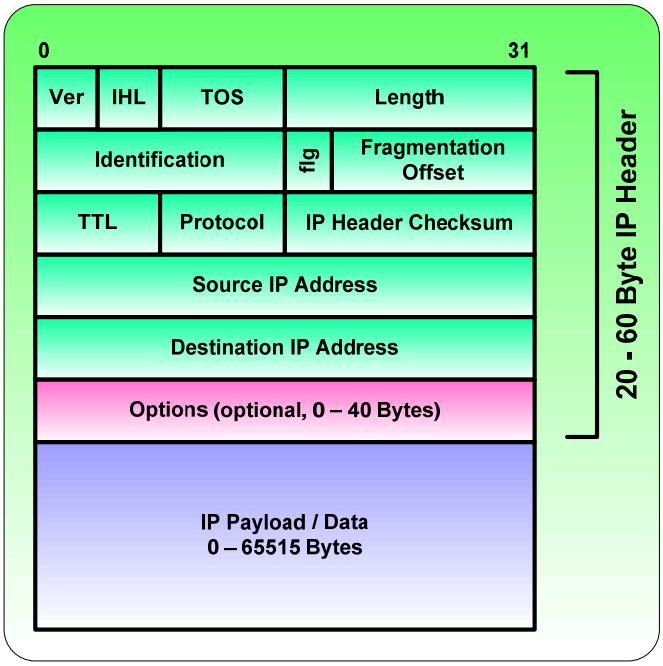

Figure 5. IP Packet Structure

Point-to-Point Protocol (PPP)

Since the link between two telemetry transceivers form a logical point-to-point link, a protocol optimized for these two-node networks would be appropriate. The PPP specified in IETF RFC 1661 provides a well-accepted standard for exactly this use case. PPP provides an efficient mechanism for carrying one or more network-layer protocols including IP, IPX, AppleTalk, and NetBIOS. PPP also provides authentication and link control mechanisms.

When only carrying IP traffic, PPP is similar to the IP forwarding approach previously presented. The encoding overhead is likely slightly worse (depending on the framing method used by the particular IP forwarding transceivers). However, this is likely outweighed by the benefit of using a standards-based approach that allows interoperability between multiple vendors' transceivers.

Application-Level Proxy

All of the approaches presented have focused on how to encapsulate the network packets that transit the network telemetry link. Adding an application-level proxy to one of these approaches provides the potential for additional network link efficiency gains. These proxies are used at the edges of the network telemetry link to repackage packets that are destined for the network telemetry link.

One method for this repackaging combines data from multiple incoming packets on the near-end network in a predictable fashion so that the original packets can be regenerated on the far-end network. Since the packets transiting the network telemetry link are larger on average than if the

proxies were not present, the average overhead from Ethernet and/or IP headers is reduced. This overhead reduction is most pronounced for applications that have significant percentages of small packets.

Proxies that also adapt the IP payload can potentially achieve larger gains in efficiency. For example, a proxy that selects individual requested samples from acquired data packets before sending to the far-end network saves from sending the entire packet through the network telemetry link. Consider the case of *TmNSDataMessages* used in LTC Data Delivery. Figure 6 shows a Wireshark packet decode of a *TmNSDataMessage* that illustrates a typical collection of packages. The *TmNSDataMessage* forms the payload of the UDP datagram. A typical *TmNSDataMessage* contains a large number (10s to 100s) of measurement samples to balance network efficiency against transport latency requirements. Consequently, an end application that is only interested in samples of a single measurement consumes available throughput of the network telemetry link with the "overhead" of all of the other measurements contained in the *TmNSDataMessage*. A proxy that creates new *TmNSDataMessages* with only the requested measurements increases the achievable application-level throughput, or goodput [1], through the constrained network telemetry link.

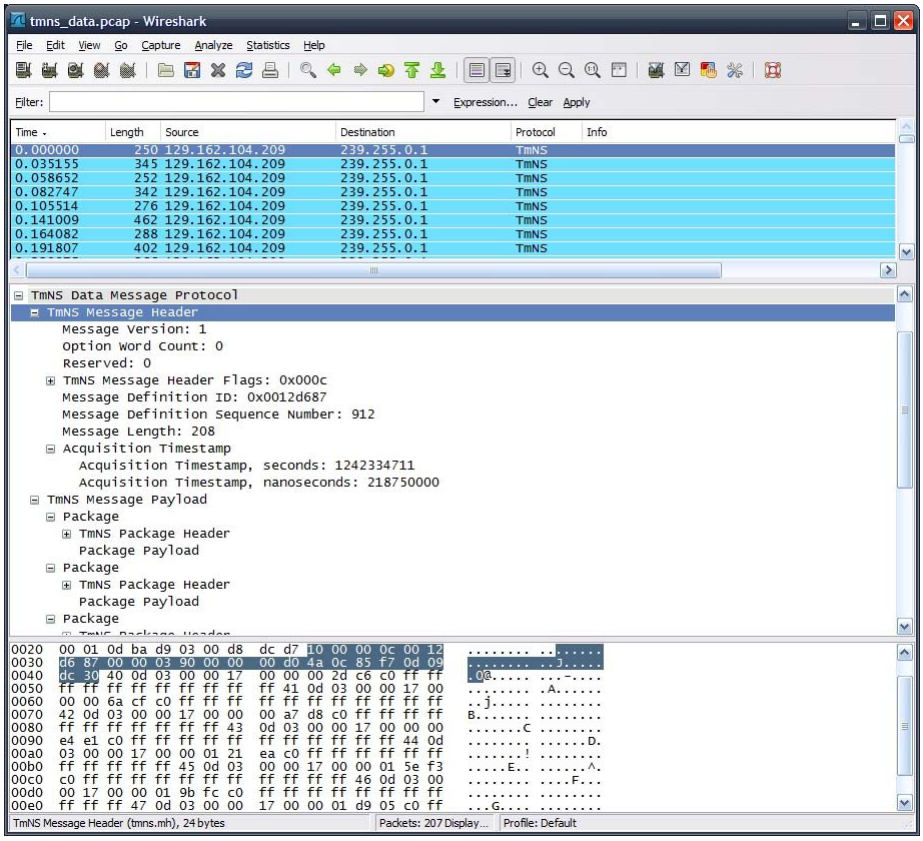

Figure 6. TmNSDataMessage Packet Decode

Besides combining the information from smaller packets into larger combined packets, the proxy must add information to allow the data to be separated back into the original shape on the far-end

network. The amount of information depends on the type of data combined, whether data is combined across sources or destinations, and whether multiple types of data (or applications) are combined. The type of combining and encoding must be weighed against the traffic statistics to determine how much network overhead reduction will be achieved.

PREDICTING NETWORK LINK THROUGHPUT

The network telemetry link throughput for each approach can be compared by applying the traffic statistics. Table 1 shows the amount of data that has to be transmitted for each approach expressed as a relative percentage of the IP forwarding approach. The analysis uses a representative data set of network traffic for each traffic category and calculates the total number of bytes required to be transferred for each of Ethernet bridging, IP forwarding, and PPP. The SNMP traffic benefits the most from the reduction of packetizing overhead due to the predominance of small packets. If all of the network traffic transiting the network telemetry link was SNMP, the Ethernet bridging approach would require 23% more capacity to achieve the same application-level throughput.

Table 1. Relative Comparison of Network Telemetry Link Data

	Ethernet	PPP	IP
Acquisition	103.1%	100.7%	100.0%
FTP	101.6%	100.4%	100.0%
SNMP	123.3%	105.2%	100.0%

Evaluating the application-level proxy approach first requires an assumption of what percentage of the incoming acquired data is selected for transit over the network telemetry link. This would vary depending on the size of the network telemetry link when compared to the total acquired network traffic data rate on the test article. Based on the TmNS shared 20 Mbps rfNET link and test article data rates in the hundreds of Mbps range, an assumption that the proxy selects 10% on average of the incoming data appears to be reasonable.

Table 2 shows the reduction in data required by using a proxy that selected 10% of each packet in the representative data set. The first row compares the Ethernet bridging, PPP, and IP forwarding approaches with the proxy as a percentage of the data required for IP forwarding without a proxy. Not surprisingly, the proxy reducing the acquisition data payload by a factor of 10 has significantly more impact on the needed throughput than which type of network telemetry link approach is used. Looking only at this statistic could wrongly lead to the conclusion that the network telemetry link approach choice has little impact. The second row shows that the Ethernet bridging approach requires 23.9% more network link capacity than the IP forwarding approach. Considering that the network telemetry link will likely be highly utilized due to spectrum limitations relative to available data on the test article, the impact of the Ethernet headers becomes a significant factor.

Table 2. Relative Comparison Using Proxy

	Ethernet	PPP	IP
Acquisition Absolute	16.3%	13.8%	13.1%
Acquisition Relative	123.9%	105.3%	100.0%

The results for the application-level proxy match closely to the SNMP traffic due to the similar average packet size. The original mix of acquisition data presented in Table 1 showed a much lower sensitivity to network telemetry link approach. However, this is to be expected when the packet size distribution in Figure 1 is reconsidered. There are a sufficient number of large packets to minimize the impact of the Ethernet headers. However, if a particular use case had only transported the smallest 25% of those packets down the link, the impact of the Ethernet headers again becomes significant as shown in Table 3.

Table 3. Relative Comparison of Acquisition Small Frames

	Ethernet	PPP	IP
Acquisition Small Frames	119.9%	104.4%	100.0%

CONCLUSION

The results presented here indicate that there are throughput performance differences between the four network telemetry link approaches. These impacts become increasingly significant as the average packet size transiting the link becomes smaller. Small packet sizes can arise from various traffic types including system management (SNMP), live acquisition data, and down-selected live or recorded acquisition data. Due to this, it is important to choose approaches that minimize the impact of small packet sizes. Thus, application-level proxies should be used with an emphasis not only on down-selecting data, but also, whenever possible, recombining the down-selected data into larger packets. In addition, Ethernet bridging approaches should be avoided in favor of PPP or IP forwarding approaches to maximize efficiency with remaining inevitable small packet traffic.

REFERENCES

1. Goodput definition, Wikipedia, http://en.wikipedia.org/wiki/Goodput