# Real World Experiences with Network Centric Telemetry Systems

Gary Thom, Richard Hoffman

GDP Space Systems, Horsham, PA, USA

**Abstract**: As telemetry ground stations are finally making the move toward network centric architectures, it is worth considering the lessons learned over the previous 10 years of designing, installing, troubleshooting and optimizing telemetry data distribution over IP networks. This paper discusses some of the architectural decisions to be made and some of the pitfalls to avoid in developing the next generation of networked telemetry ground stations. Critical issues such as latency, efficiency, data loss and Quality of Service are addressed, as well as techniques for troubleshooting these problems.

**Keywords**: Internet, IP, TCP, UDP, TMoIP, network, PCM, Latency, packet loss

## 1. Introduction

Companies like GDP Space Systems have been sending PCM data over packet switched networks for over a decade. In the beginning, there was very little interest in this new method of transporting PCM data. There were microwave links, fiber optic links, coax cable, and matrix switches. They all served the industry very well for many years.

During this time, IP networks have grown in usage, capacity and capability. In the data world, they have become ubiquitous, allowing data to be distributed worldwide in the blink of an eye. Reliability and redundancy are built in and provide guaranteed delivery of an infinite range of data.

The first non-traditional data type that began to move to IP networks was voice. The low cost of data transport over IP networks drove a cottage industry in toll bypass and low cost international voice traffic over IP networks. This was followed closely by video when video conferencing moved from circuit switched networks to the packet based IP networks. This was followed by the cable TV industry changing from RF video distribution over cable to packet based digital video transmission over coax and fiber.

This set the stage for sending PCM data over IP networks or Telemetry over IP (TMoIP).

## 2. Why do we want to use TMoIP?

The motivation for moving to TMoIP was twofold: first, to find cost effective PCM data distribution and second, to provide reliable and robust PCM data distribution regardless of the destination. The global explosion of IP networking has provided a built in infrastructure with access to the most remote destinations. A wide variety of transport mechanisms for IP traffic provides ubiquitous connectivity, whether twisted pair, fiber optic cable, microwave links, satellite links, analog modems and cell phones, IP connectivity was everywhere.

This ubiquity and global deployment drove down the cost of networking components such as routers and switches. It provides dynamic routing and redundant paths, improving reliability and fault tolerance. The insatiable appetite for more data has resulted in ever increasing bandwidth availability.

The result is a reliable, cost effective infrastructure for PCM data distribution, whether on private IP networks or globally via the public internet.

A side effect of the move to IP is that with very little effort, the TMoIP gateway device could provide frame aligned packets of data. This proved to be extremely useful for driving a growing array of software decoms. These are software packages that run on standard computers that are capable of decommutating, processing and displaying telemetry data. By providing frame aligned packets of PCM data, the hardware frame synchronizer and serial to parallel converter in traditional hardware decoms could be eliminated. These software decoms provide a low cost alternative to purpose built hardware decoms for application where data rates and processing loads could be managed by a general purpose CPU.

Another side effect of the move to IP is that these packets of PCM data could very easily be captured by a computer and stored to disk in industry standard formats such as IRIG 106 Chapter 10. This provides some efficiency by converting all PCM data to IP at the edge of the network where it could then be easily distributed, decommutated or recorded.

## 3. Sending PCM data over IP

**3.1 Overhead -** When TMoIP data is transmitted on the network, there are several levels of protocol that add header and trailer data to the PCM payload data. Each layer of the protocol stack provides additional functionality for addressing and routing packets between the source and destination. For TMoIP applications, these protocols include Ethernet, Internet Protocol (IP), User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) and finally, the IRIG 218-10 TMoIP Protocol.

All of these protocol specific headers and trailers are overhead which increases the effective data rate needed to send the PCM data over the network. This is shown for IPv4 in Table 1.

**Table 1 - IPv4 Overhead (in bytes)**

| | |
|---|---|
| Ethernet Frame Preamble, SOF and Inter-packet Gap | 8 |
| Ethernet Header | 18 |
| IPv4 Header | 20 |
| UDP/TCP Header | 8 / 20 |
| IRIG 218 TMoIP Header | 4 |
| Total Overhead | 70 / 82 |
| Maximum Payload Size | 1468 / 1456 |
| Minimum Overhead | 4.7% / 5.6% |

These are the minimum header sizes and they do not contain any optional bits such as the VLAN tag, which will further increase the header sizes. Obviously, payload size has a tremendous impact on the efficiency of TMoIP. The Ethernet packet payload is limited to the Maximum Transmission Unit (MTU). The maximum MTU is 1500 bytes, however, this may be set lower in some networks. The IP header, UDP/TCP header and TMoIP header size must be subtracted from the MTU in order to determine how much payload the UDP or TCP packets can carry.

The IPv6 overhead is shown in Table 2.

**Table 2 – IPv6 Overhead (in bytes)**

| | |
|---|---|
| Ethernet Frame Preamble, SOF and Inter-packet Gap | 8 |
| Ethernet Header | 18 |
| IPv6 Header | 40 |
| UDP/TCP Header | 8 / 20 |
| IRIG 218 TMoIP Header | 4 |
| Total Overhead | 90 / 102 |
| Maximum Payload Size | 1448 / 1436 |
| Minimum Overhead | 6.2% / 7.1% |

The payload size is one parameter that we have control over in producing TMoIP packets. Making the payload larger will reduce the overhead percentage and increase the transmission efficiency at the expense of latency. Conversely, making the payload smaller will increase the overhead percentage and decrease the transmission efficiency, but the latency will be reduced.

Jumbo Frames allow the Ethernet packet payload to expand beyond the 1500 byte limit up to 9000 bytes. However, these are illegal Ethernet packets and network equipment must specifically support the use of Jumbo Frames. If they do not, packets may be truncated or chopped up into smaller packets as they pass through various network equipment.

**3.2 TCP vs UDP -** One of the first choices to make in sending PCM data over an IP networks is the use of TCP or UDP transport mechanism. TCP provides a guaranteed delivery service that assures that the data being sent will arrive at the destination. This is achieved by implementing an acknowledgment and retransmission system. Data must be buffered in the transmitter and receiver to allow retransmission of non-acknowledged data. This buffering leads to increased latency as the buffer must be large enough and contain enough data so that it will not run dry while waiting for retransmission of missing data.

A second feature of TCP is that it provides a byte stream. You would think that this would be useful in sending a stream of PCM data; however, TCP has some issues that make parsing a stream a little more difficult. Data sent to a TCP socket in the sending device does not necessarily end up in a single packet. The byte stream is divided into packets based on the TCP window size, timeouts and other criteria. So, the start of a TMoIP packet may no longer be aligned with the start of a TCP packet. This requires the TCP byte stream to be parsed byte-by-byte for the TMoIP packet header.

TCP service only provides a point-to-point connection. It requires communications in both directions in order to set up the connection even if you are only transmitting data in one direction. There is a connection setup process (SYN) that takes place and that must be completed successfully before any data is transmitted. This is not an issue in typical IP network environments, but there are some applications where bidirectional communications is not available. Some examples are: 1) RF transmission of TMoIP data or 2) transmission of TMoIP traffic through an “optical diode” for security reasons.

UDP does not provide a guaranteed delivery service. There is no connection process. Data is fire and forget. There is no acknowledgement that packets have been received and no retransmission of lost packets. The benefit of this service is that it is very low latency. No buffering is required. The down side is that lost packets are lost forever, however, in modern networks that are not congested there is very little or no packet loss.

The second benefit of UDP is that it is a datagram service. That is, the data sent to a UDP socket in the sending device is sent intact within a single UDP packet (assuming the size does not exceed the MTU limit). If the sending device aligns a TMoIP header with the start of the UDP packet, the receiving device will receive that TMoIP header at the beginning of the received data. This significantly simplifies parsing of the stream.

While the quick reaction may be to use TCP for TMoIP applications because of the guaranteed delivery of error free data, most practical applications use UDP for the following reasons:

1) Low latency
2) Simplified parsing of datagrams
3) Availability of multicast service
4) Does not require bi-directional connections

**3.3 Multicast vs Unicast –** UDP can operate in two ways: unicast or multicast. Unicast allows a packet to be sent from one sender to one specific receiver. This is a very simple, very straight forward method of passing data. For many TMoIP applications, this is the normal mode of distributing PCM data. The sender decides on the destination of the packets.

UDP also provides a multicast service where one sender can send to multiple receivers. This is a very effective and efficient mechanism to provide a similar capability as a non-blocking matrix switch where one input can be connected to multiple outputs. But a second feature of multicast TMoIP service is that senders can send without knowing the destination. This is very effective in those cases where multiple PCM streams need to be made available to everyone and the receivers will decide which PCM streams they want to receive.

**3.4 TMoIP Applications -** There are two primary applications for the use of TMoIP. These are PCM-to-PCM data distribution over IP networks and PCM-to-Computer distribution over IP networks.

In the PCM-to-PCM data distribution application, shown in Figure 1, we are using the IP network to replace traditional PCM transport mechanisms such as copper cable (coax or twisted pair) links, microwave links, fiber optic cable links and matrix switches. In these applications, we start with PCM as serial data and clock signals. This is applied to the TMoIP Gateway which converts the serial data to parallel, formats the data into TMoIP packets and then outputs them over the network using TCP or UDP transport. At the receiving end, the TMoIP Gateway receives the TMoIP packets, strips off the various packet headers and converts the parallel data back to serial data and clock signals.


**Figure 1 – PCM-to-PCM Data Distribution**

In the PCM-to-Computer distribution application, shown in Figure 2, we are using the IP network as a convenient input interface to a computer. The computer would typically be running a recording and/or decommutation application program. This usage has gained a lot of traction where the data rates, numbers of PCM streams and processing requirements can be supported by the processing power of the computer CPU. In addition, an Ethernet input provides a very simple and generic data interface to an application which eliminates the need for special hardware interfaces and operating specific device drivers.

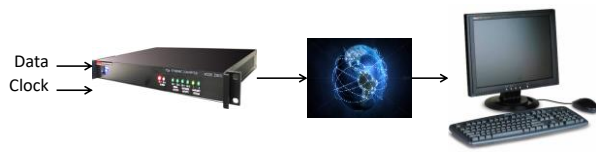



**Figure 2 – PCM-to-Computer Data Distribution**

## 4. TMoIP Problems and Issues

**4.1 Latency -** Latency is the delay from the time a bit enters the PCM-to-Ethernet device until that same bit leaves the Ethernet-to-PCM device. Everyone wants zero latency; however, some latency is unavoidable. The good news is that to a certain extent it is controllable. End-to-end latency is made up of several components including Processing Delays, Packetization Delays, Network Delays and Receive Buffering Delays

Device Processing Delays are vendor specific and depend on serial-to-parallel / parallel-to-serial conversion, DMA buffering, packet processing and network stack delays. Efficient design will minimize these delays, but they are out of the control of the user.

Packetization Delays are often user controllable, but there is a trade-off. Data is buffered while enough data to fill a packet is received. Once the packet is filled, the packet can be sent. So making packets smaller will minimize that delay. This comes at the expense of efficiency because the smaller the packets, the larger the impact of packet header overhead, so the larger the effective data rate will be.

Network delays are the time it takes a packet to traverse the network from source device to destination device. This delay is not only made up of the travel time of the electrical signals, but also the delays through the intervening network devices (switches, routers, etc.), each of which may implement its own queueing and buffering schemes.

Receive Buffering Delays are similar to Packetization Delays in that the entire packet must be received before any of the data can be processed. This is because a Frame Check Sequence at the end of the Ethernet packet must be checked before the packet is released to the waiting processes. Additional buffering may also be required at the receiving end to compensate for network jitter.

As a result of all of these components it is very difficult to say exactly what the latency will be. However, assuming the network is well designed and not in a congested state, all of the device and network delays should be fairly constant, allowing the user to control latency by specifying the packet size and jitter buffering.

**4.2 Skew -** Skew is the channel-to-channel difference in the latencies. If this difference is great, then PCM events that occur at the same time at the source will no longer occur at the same time at the destination. Like latency, users always want zero skew, and like latency, some skew is unavoidable. Supposed that you have two PCM data streams, one at 5 Mbps and one at 32 Kbps. Supposed that you also want to maximize efficiency so you use large packets for both streams. There will be a significant difference in the latency of the two streams resulting in a very large skew. Now a user can attempt to compensate for this manually by adjusting the packet size of each stream, but this becomes difficult if the data rates change

often, or especially if they are not know at the time the link is configured. However, there are two automatic ways to minimize skew.

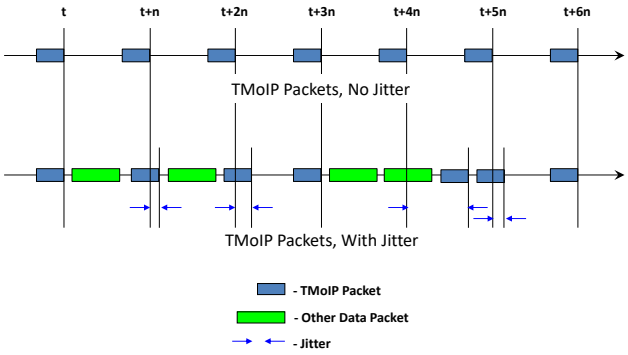
One method is to timestamp each TMoIP packet at the source, to synchronize all TMoIP devices to a common time base, to identify the largest latency of all channels of interest, to use that latency as an offset to the current time for establishing a playout time, and then to playout the TMoIP data when the timestamp matches the playout time.

Another approach is to attempt to control all of the latencies to a common value. Assuming the device and network latencies are fixed, this amounts to controlling packet size and buffering for each channel of interest.

**4.3 Jitter -** PCM data has a constant, continuous data rate. Every bit time, a new bit of information is available and the bit clock is generally at a constant stable rate. Once the PCM data is packetized and sent onto the network, this is no longer the case. The packets containing the PCM data are sent at higher data rates in short bursts. This is shown at the top of Figure 10. If there is no other traffic on the network, then the packets can be sent at a regular rate and will reach the destination at that same rate, making it easy to convert back to a constant bit rate serial PCM output.

But when other traffic is present on the network, the TMoIP packets may not be able to be sent at a regular rate. As a result, when packets arrive at the destination, they have jitter as shown at the bottom of Figure 3.

**Figure 3 - Packet Arrival Time Jitter**




The problem with jitter is that when data is needed for the parallel to serial conversion process, the next packet may not yet have been received. The serial PCM stream is a continuous stream of data. Any break in that stream will result in loss of bit sync or decom lock and errors in the resulting data.

**4.4 Packet Loss -** Packet Loss is when packets are sent over the network and they do not reach the destination. There are many reasons for packet loss which will be discussed in this section.

The first reason is physical. In this case, there is a bad, damaged or out of spec cable. This would also include damaged connectors which do not seat properly and damaged or faulty network equipment. The result may be bit errors in the data transmission. The Ethernet layer contains a CRC check, which if it fails causes the entire packet to be dropped.

The next common cause is incorrectly configured Ethernet data rate and duplex settings. This is very commonly misunderstood. If two devices are connected together using an Ethernet cable, both devices must be configured the same way. The auto settings for the network interface means auto-negotiate the data rate and the duplex. If both devices are set to auto-negotiate, then they each advertise their supported data rates and duplex and then select the highest common rate. Today, this tends to be pretty reliable method of configuring the interfaces. If both devices are manually configured, they must be configured the same. This seems obvious, but is the cause of many networking problems. If the data rates are set differently, there will be no communications between the devices. If the duplex is set differently, there will be an apparent connection, Pings may work properly, however, the communications will be unreliable and will result in dropped packets due to collisions. However, if one device is set to auto-negotiate and the other device is set manually, the negotiations will fail. In that case, the auto configured device will default to 10Mbps and Half Duplex. If the manually configured device is set to Full Duplex, a duplex mismatch will occur with the resulting dropped packets. Some devices will provide auto-detection for the data rate when negotiation fails. This will provide a data rate match. However, there is no auto-detection for the duplex.

These first two causes of packet loss are easily corrected. The next cause is a little more difficult. That is packet loss due to congestion. Congestion is the condition where more data is passing through a network device than the device is able to handle. In this case, the device will discard packets in order to prevent buffer overflow.

Packet loss not only affects the integrity of the output of the Ethernet-to-PCM data stream, it can also affect the output clock rate. Typically, the Ethernet-to-PCM device will monitor buffer fullness in order to provide fine control of the output clock to prevent buffer overflow or underflow. If packets are dropped, this can throw that mechanism out of whack resulting in abrupt changes in the output clock rate. To prevent this, the missing packets must be accounted for in the buffer measurement process.

## 5. Architectural Considerations

This section describes some of the real-world considerations in designing a TMoIP network.

**5.1 Private vs Shared networks -** A Private network is one that is specifically allocated to transmitting PCM data. No other data flows are present. A Shared network is shared between PCM data and other data types. If the purpose is for casual quick look of some PCM stream and the data rate is modest, a Shared network may be perfectly suitable. In other cases, a Private network may not be possible because the public internet may need to be used. Obviously, in these cases, you will have less control over

the quality of service and reliability of the PCM data transmission.

However, if the transmission of PCM data is mission critical, it is recommended that a Private network be built. This will isolate the PCM traffic from any other less critical but potentially disruptive traffic. If a completely physically private network is not possible, it may be possible to approximate a Private network by using Virtual LANs or MPLS tunnels.

**5.2 Data vs Management networks -** Separation of Data and Management networks provides two advantages in TMoIP systems. First, it removes the computer systems that perform the management (setup, configuration and status) function from the same network that is being used to the TMoIP data. As mentioned earlier, this has the benefit of removing a lot of operating system generated network traffic from the data network. This traffic has the potential to increase network jitter resulting in higher latencies.

The second advantage is security. Often, the TMoIP network will be carrying classified data which must be segregated from other data. Eliminating computer systems from that network reduces vulnerability and simplifies the Information Assurance certification of the network.

**5.3 Quality of Service Functions -** There are a variety of Quality of Service mechanisms that can be used to improve the performance of the TMoIP network.

**5.3.1 Traffic Classes -** Both IPv4 and IPv6 provide fields in their headers for marking the packets with a Class of Service. In IPv4, this is the Type of Service (TOS) field also called the Differentiated Services Code Point (DSCP) field. In IPv6, it is the Traffic Class field. These fields are used to identify specific traffic flows that should have special handling applied as the packets pass through the network. For example, all normal traffic could be given a Class of 0, Voice over IP traffic could be given a Class of 6 and Telemetry over IP traffic could be given a Class of 7. In this way, Voice and Telemetry traffic can be given special handling.

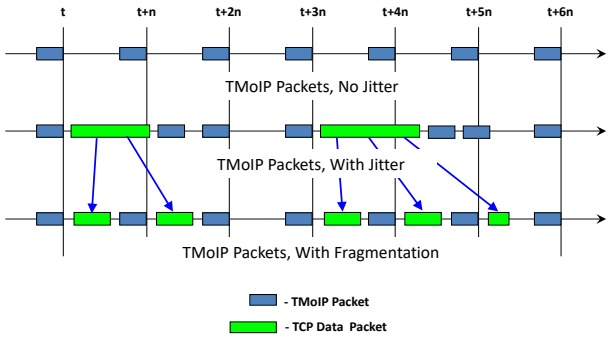
Setting the traffic class alone, does not provide any improved quality of service. You must also configure the routers in the network to treat the classes differently. For example, traffic classes 6 and 7 can be given Expedited Forwarding treatment and class 0 can be given Assured Forwarding. Expedited Forwarding puts those packets into very short, high priority queues. Whereas Assured Forwarding packets are put into longer, low priority queues. The router will output packets from the high priority queues before outputting packets from the low priority queues. This helps to minimize latency and jitter. It also helps to avoid dropped packets as network congestion grows. Assured Forwarding is typically used for low priority TCP packets, knowing that these packets will eventually be retransmitted if dropped.

There are many other features that can be programmed in the routers that determine how different classes of service are treated. There are different queueing algorithms: First-in-First Out, Weighted Fair Queueing, and Class Based Weighted Fair Queueing. These are used for prioritizing the traffic flows. Next, there are Congestion Avoidance tools that discard packets in order to avoid congestion. These include: Random Early Detection and Weighted Random Early Detection. In both cases, impending network congestion is detected when queues are nearing their full level. When this is detected, packets are dropped. When packets are dropped, the TCP packet senders do two things. First, they delay their retransmission as more packets are dropped, and second, they decrease their acknowledgement window sending fewer packets between acknowledgements. Both of these actions work to reduce the rate of low priority, guaranteed delivery data being sent into the network. If there are no TCP packets to be dropped, then low priority UDP packets will be dropped. These will not be retransmitted and therefore will be lost.

For Quality of Service mechanisms to be effective there should be an end-to-end policy which is implemented in each network device that the data will transit through. If not, there is the potential that the desired handling will fall apart in the unprotected segments, and quality of service will not be maintained.

**5.3.2 Packet Fragmentation and Interleaving -** For those cases where the TMoIP data cannot be separated from common Ethernet data, Packet Fragmentation and Interleaving can help improve performance. Packet fragmentation and interleaving is a mechanism that helps to reduce the jitter of high priority, low latency real time data streams such as TMoIP. It works in the routers by chopping up large TCP packets into smaller pieces. This prevents the UDP traffic from having to wait while large packets are being sent, minimizing the packet arrival time jitter. This does not affect the TCP traffic which just spreads the byte stream over more packets. This is shown in Figure 4.

**Figure 4 - Packet Fragmentation**




**5.3.3 Bandwidth Considerations -** It is critical to assure that the volume of telemetry data to be transmitted over the TMoIP network does not exceed the capacity of the network. This is easier if the network is a TMoIP-only network and is not shared with other traffic. But even in this case, care must be taken to account for the packetization overhead when determining the aggregate bandwidth requirements. Even after accounting for the overhead, it is important to limit traffic to about 85% of

the wire speed to account for transmitter and receiver recovery time and intermediate network equipment processing loads. Modern network devices are designed to operate at wire speeds, however, they often have packet processing limitations. Small packets at very high data rates may overload the devices ability to process and pass data.

In those cases where the network is shared with other Ethernet traffic, it is difficult to estimate the impact of the non-TMoIP data. This data tends to be bursty and unrestricted in data rate. Generic Ethernet traffic such as HTTP, FTP, SMTP, POP is very intermittent and unpredictable. This type of traffic can cause serious issues with the reliability of TMoIP traffic if the aggregate data rates are approaching wire speeds or the data rate limitations of intermediate network equipment.

## 6. Troubleshooting

There are two primary categories of problems that may affect TMoIP networks. The first category is connectivity problems. In this case, packets sent from the source do not reach the destination. There are several causes of connectivity problems. In these cases, a route does not exist between the source and the destination. This may be the result of a physical break in the network path, either due to cabling or architecture. This may be the result of mis-configuration of firewall or network address translation (NAT) functions which make destinations unreachable even though the physical connection is possible. Similarly, having the source and destination on different VLANs with no intervening VLAN router will make the destination unreachable. And finally, if the traffic is using multicast, multicast traffic may be blocked by routers or switches.

To troubleshoot these types of problems, utilities such as Ping or Tracert can be useful. Some TMoIP devices provide these utilities within the device. If the device does not, then a laptop can be substituted for the source device in order to use the utilities. Ping can then be used to walk from source to each intermediate network device in order to determine where the route is blocked. At that point the configuration of that network device can be investigated. There may be cases where the Ping is successful, but there still is not connectivity from source to destination for TMoIP traffic. In this case, it is likely that a Firewall is blocking the UDP traffic but passing ICMP traffic which is used for the Ping utility. Additionally, a switch or router may be blocking multicast traffic, but passes UDP and ICMP data.

The second category of problems that affect TMoIP networks is dropped packets. In this case, there is connectivity, and some data gets from source to destination, however, some packets are lost which results in data errors at the destination. This may be the result of a physical problem such as a damaged cable, connector or electrical interface. Another possible cause of packet loss is duplex mismatch which was previously discussed. Network congestion may also result in dropped packets, and is also discussed in a previous section. It may also result in excessive packet jitter which in some cases may cause buffer underflow in the Ethernet-to-PCM device which also results in data loss. And finally, exceeding the network data rate will result in lost packets. While this seems obvious, it may not always be easy to identify. This is because intervening network segments or connections may be operating at a lower data rate than expected. For example an intermediate segment may be configured for 10Mbps or a wide area network connection may be going through a low data rate telecom connection.

These conditions can cause very intermittent errors which can occur anywhere in the network chain and may be very difficult to track down. Troubleshooting these problems is equally difficult. Checking duplex and data rate settings along the network path is an easy task. Checking for congestion within network devices may be done by checking the device statistics. While checking these conditions in private networks may be relatively easy, it may be much more difficult in public networks where you do not have control or access to intermediate network devices. Physical problems are very difficult to find. A cable may be crushed or bent in such a way that the impedance is affected. A break in ground signal may affect common mode noise rejection. In improperly polished or crushed fiber optic cable may decrease signal quality. Bent pins and pushed out pins may result in intermittent connections. These will all lead to dropped packets and can only be found through careful examination of cables, connections and equipment in the network path.

## 7. Conclusions

IP networks provide a very efficient and effective means of distributing real-time low latency PCM data as long as attention is paid to the network architecture, packetization and buffering.

The wider use of TMoIP requires telemetry engineers to become proficient in network configuration and troubleshooting.

Currently, IRIG 218 [1] attempts to standardize the TMoIP protocol. However, in many cases, the requirements are not explicitly stated, resulting in spotty acceptance and limited interoperability. IRIG 218 will be soon undergoing revision and the hope is that some of this real world experience can be taken into account in producing a more widely accepted standard.

## 8. References

[1] Telecommunications and Timing Group: “IRIG 218-10 Telemetry Transmission Over Internet Protocol (TMoIP) Standard”, Range Commanders Council, 2010.