MEASURING LATENCY IN IRIDIUM SATELLITE CONSTELLATION DATA SERVICES

233

Margaret M. McMahon, Ph. D.
Computer Science Department, US Naval Academy
572 M Holloway Rd, Stop 9F
Annapolis, MD 21402
mmmcmahn@usna.edu

Robert Rathburn
ISDI TECHNOLOGIES, INC.
1600 Kapiolani Blvd, Suite 1100
Honolulu, Hawaii 96814
rrathburn@isdi-hi.com

Abstract

The use of Satellite Communications (SATCOM) has become essential to operations in both Afghanistan and Iraq. In particular, the Iridium satellite constellation has demonstrated its usefulness and flexibility. It has had significant impact on how operations are conducted.

Iridium provides users both voice and data services. There are two approaches to sending data over the Iridium network: a circuit-switched data service and message-switched data service. Use of the circuit-switched data service requires the overhead of a setup in the same manner as a voice call. There are two message-switched data services: Short Burst Data (SBD) and Short Message Service (SMS). Our testing involves the SBD message-switched data service, which is optimized for high capacity and efficiency when sending small amounts of data through the Iridium network.

While there are many users for these services, little information can be found about their actual performance in fielded systems. The best sources of information are specifications or modeling of the performance of these services. We perceived that understanding the underlying network performance was needed, and we established experiments to capture that performance. We present results from both circuit-switched data service and message-switched SBD data service. We also address insights into the use of these services.

1 Relevance to Command and Control

Effective command and control in regions without an established or accessible communication infrastructure relies on the communication paths provided by geosynchronous or satellite constellations. When a network application uses a satellite, there are expectations of latency. Network-centric application developers need to understand that underlying performance; and take into account operational characteristics of commercial services such as satellite call capacity and call setup times. With Low Earth Orbit (LEO) constellations such as Iridium, there are also considerations associated with look angles to satellites and revisit and coverage times of the constellation. Contrary to common understanding, Iridium service does not cover the whole globe at all times.

An important part of Command and Control systems is the timely delivery and fusion of data from multiple sensors that often have overlapping areas of coverage. The Tactical Component Network (TCN®) technology transparently integrates sensor and communications suites with distributed network applications. TCN has a local component, called the TCN Local Network that handles time-critical, peer-to-peer applications, and a wide-area capability called the TCN Global Network. The global TCN uses a hub-and-spoke architecture to allow geographically dispersed systems to share data, and to interact with applications resident on the hub.

The Iridium constellation is used as one type of spoke in the hub-and-spoke architecture of the global version of the TCN. Iridium was employed during US Navy 7th Fleet exercises, where TCN was used to integrate sensor data from multiple ships in a Beyond Line Of Site (BLOS) operational environment. At the end of each spoke into the hub was a Tactical Component Network (TCN) node, typically consisting of a local area network (LAN), multiple processors, cryptographic equipment and various peripherals. Some data provider nodes connected TCN to sensors such as the AN/SPS-48 radar on USS Essex, and others such as the USS Guardian provided mine information to other users. Currently, global TCN data is transmitted via circuit switched method via an encrypted Iridium data link. The message-based SBD service is being investigated as an alternative data transmission method for applications not requiring continuous connectivity.

2 Introduction

This paper addresses the methods and experiments used to gain insight into the performance of spokes when they are implemented by the Iridium constellation.

Our approach was to measure the actual latencies for these data services. In separate experiments, circuit-switched data calls and SBD message-switched data transmission latencies were quantified. For the circuit-switched experiments, a custom client-server application was built and instrumented. The SBD types of transmissions analyzed were: Mobile Originated (MO), Mobile Terminated (MT) and Full Duplex.

3 Related Work

In [1], the authors describe an application of the Iridium network to support communication needs of passenger, flight and cabin crews for voice, data and paging. Voice latency was estimated to be between 270-390 milliseconds (ms). Effective channel throughput delay for data (1024 bit message) is estimated between 427 ms and 1.7 seconds. Simulation of Iridium satellite networks performed by [5] estimated an upper limit of the average end-to-end delay to be 210 ms when passing through twelve satellites. These studies were based on assumptions, rather than measurements.

4 Background

The data transfer rate for Circuit-Switched Data (CSD) is 300 Bytes/sec and for SBD is 125 Bytes/sec. A CSD data requires that the satellite acquisition and call setup has been completed. However, an SBD transfer requires only that satellite acquisition has completed. This potentially makes SBD a better choice for Low Probability of Intercept (LPI) / Low Probability of Detection (LPD) applications [3].

Given that satellite acquisition has occurred, SBD is able to transfer 100 Bytes in approximately 1 second. Each SBD call can transfer a maximum of to 1960 Bytes from a mobile system to a gateway, and a maximum of 1,890 Bytes from gateway to a mobile system [3].

5 The Experiments

5.1 Circuit-Switched Data Service Experiments

The CSD experiments used the hub-and-spoke architecture of Global TCN. The hub is similar to the telephone central switching office. In addition to facilitating communications between dissimilar systems, it contains a real-time repository of historical and current data. Combining these features enables the hub to be the central point host for network-centric applications.

Although the hub is a centralized concept, its functions can be replicated to prevent having a single point of failure. Spokes, as well as the hub, can be redundant to maintain communication capability. A backup hub maintains the current hub configurations and can prevent disruption of the services and connections.

The experiments for circuit switched applications were conducted using a hub located in Laurel, MD, and clients were located in Annapolis, MD. The direct distance is 21 statute miles (42 miles round trip). A combination of the Iridium constellation and land-line created a spoke into the hub. The Iridium gateway in Phoenix, AZ, was used. The Iridium satellite network carried traffic from the client laptop to the ground station in Arizona (2008 miles). The trip to the hub in Laurel was via land-line (1989 miles).

To better understand the underlying network performance, we eliminated the global TCN application in our experiments and used representative data streams.

5.2 Message-Switched Experiments

There are two types of message-switched services offered in the Iridium constellation: SBD and SMS. The SBD service allows the use of a modem, supporting communicating applications through a method similar to e-mail [3]. The SMS service is a text-messaging service for sending messages from one cell phone to another [4].

The message-switched experiments used the SBD service with both Mobile Originated (MO) Transmissions and Mobile Terminated (MT) Transmissions. These types of transmissions were operated simultaneously in full-duplex mode and are described in the next sections. The mobile equipment was located in Honolulu, HI, and the service used the gateway in Tempe, AZ. The distance from Honolulu to

Tempe is 2915 miles. Both the sender and receiver were mobile units; there was no land-line component in the SBD experiments.

6 Short Burst Data Overview

6.1 Mobile Originated Transmissions Overview

MO transmissions originate from the mobile device and are sent to the Iridium gateway and ultimately to the destination e-mail address(es). The maximum size of a MO message is 1960 bytes (one 70 byte header segment followed by up to fourteen 135 byte data segments). The message uses the signaling path, vice a circuit switch call origination procedure, to transfer the message to the satellite. The message is then delivered to the Iridium gateway, formatted into an attachment on an e-mail message and forwarded immediately to the destination e-mail address. The mobile unit receives a status signal indicating whether the transmission was successful or not, allowing the controlling application to determine the next logical step (retry or skip). Two separate applications were required in order to facilitate the testing process: the MO Send application and the MO Receive application.

The MO Send process is the main driver of the SBD process, controlling the mobile unit and checking the transmission status of the MO. To send an MO message, the controlling application must have the message, the message length, and the 2 digit checksum (least significant 2-bytes of the summation of the entire SBD message). The transmission is executed with standard modem “AT” commands to: load the message to the mobile unit, initiate the transfer, and clear the buffer. The MO process will provide notification to the application if any of the steps is unsuccessful.

The MO Receive process is the means by which a mobile subscriber receives messages destined for its specific ID. The Receive process is not permanently connected to the server that holds e-mails intended for delivery and must poll the server to determine if e-mail(s) are addressed to its specified ID. The mobile subscriber's Receive process may be configured to poll for incoming e-mail infrequently or request e-mail at the maximum request rate. The MO Receive process functions as an e-mail client that connects directly to the e-mail server that is specified as the destination e-mail address. The client must recognize messages destined for the target e-mail address and select the messages with the correct mobile unit International Mobile Equipment Identity (IMEI) in the subject line. The IMEI serves as a serial number recognizable to the Iridium SBD system. As MO messages reach the gateway, they are immediately sent to the destination e-mail address. Additional functionality may be built into the receiving application to provide the option to either download a backlog of e-mail messages or selectively review the queue of previous messages.

6.2 Mobile Terminated Transmissions Overview

MT transmissions are SBD transmissions that originate with an e-mail message to the Iridium gateway, which is destined for the mobile unit (identified by IMEI number). The maximum size of a MT message is 1890 bytes (up to fourteen 135 byte segments). The e-mail message to the Iridium gateway must be formatted with the actual MT message as an attachment. After the message has been sent to the gateway, an acknowledgement is received to indicate whether the message has been successfully queued or not. The mobile unit is then responsible for retrieving the MT message by sending a MO message that connects to the gateway and retrieves the next available message from the queue. The message is extracted from the mobile unit through a series of standard modem "AT" commands.

The MT Send process must function as an e-mail client capable of sending and receiving messages to and from a specific e-mail account at a mobile terminate location. When a SBD message is to be sent, the MT Send process generates an e-mail message to the Iridium gateway and includes the SBD message as an attachment with the IMEI included in the subject line of the e-mail message. At the gateway a process runs every 30 seconds to gather the MT SBD messages and queue them for the individual mobile devices. An acknowledgement message is sent back to the originating e-mail address confirming whether or not the message was queued successfully for transmission. Iridium

Documentation indicates that the queue can store up to 50 messages per IMEI, however in our testing we found that the limit appeared to be 200 messages. If the queue is full, the acknowledgement will indicate an error and provide notification that the message was not queued successfully. If an error is indicated, the MT Send process can determine whether to suspend sends, resend the message or skip to a subsequent message.

There is a secondary confirmation that the message was actually transmitted successfully, however this status must be extracted from the body of e-mails that are sent as the result of an MO send (actual message or 0 byte mailbox check). According to documentation, testing, and discussions with Iridium, if the messages are queued successfully they will be transmitted successfully. We found nothing to contradict this assertion during our testing.

The MT Receive process is responsible for periodically checking the Iridium gateway queue to see if any messages are waiting to be transferred. While some additional functionality may soon be available (e.g. MT Ring Alert, which is used to notify when a MT message is awaiting transmission), the Receive process must poll the mailbox to determine if there are messages to be transmitted and identify how many messages are currently in the queue. A mailbox check is performed by sending an empty (0 byte) MO message to check if there are MT messages stored in the queue waiting to be retrieved. The mailbox check is accomplished by the AT commands to: clear MO send buffer, initiate the SBD transfer, and retrieve MT message from the buffer. The MO Send process will also trigger a mailbox check.

The Receive process can only retrieve one MT message at a time. In the process of initiating the transfer, the mobile unit will provide a receiving process status indicating whether a MT message was retrieved, the size of the message, and the number of messages still waiting in the queue to be transmitted. The Iridium gateway generates an e-mail message to the MO destination address that contains the transfer status to indicate that the MT message was successfully retrieved, which can be monitored to provide absolute verification that the MT message was transmitted successfully.

6.3 Full Duplex Overview

Testing SBD in Full Duplex mode requires traffic flowing both to and from the mobile unit. The MO Receive and MT Send processes are tightly coupled. Whenever an MO message is sent, a query is performed to detect if there are any MT messages in the queue awaiting transmission. If a message is found, the mobile unit retrieves the message and makes it available to the MT Receive process. Additionally, if the period between sending MO messages is too long, additional mailbox checks may be required between MO messages.

7 Experimental Setup

7.1 Circuit-Switched

We used a custom, instrumented, User Datagram Protocol (UDP) stateless echo client and server applications, implemented in the C programming language. The applications used blocking socket functions. Both computers ran Windows XP, and were synchronized to Universal Coordinated Time (UCT) attained from Global Positioning System (GPS) satellites. In these experiments, we used laptop computers running only the Windows XP operating system (OS) and the echo application. Communication functions were supported by the OS. Round trip time (RTT) was calculated by calculating the time elapsed by the client between sending and receiving a packet, subtracting the time between receiving and sending text at the server program.

7.2 Short Burst Data

Using SBD service requires an Iridium phone with Data Kit attachments or the Iridium 9522 L-Band Transceiver (with power supply) and an antenna to meet the needs of a mobile unit. Using a Motorola Mobile phone requires a model capable of SBD and the flash version firmware that supports SBD. The L-Band transceiver functions as a modem for accessing the Iridium Satellite network from a computer. Additionally, software setup is required to support SBD. The unit (phone or transceiver) must be provisioned and a SIM card provided by Iridium must be installed. Iridium also provides access to a software system for tracking system usage and performing simple maintenance on the SBD transmissions.

In our SBD experiments, there are transmissions between a mobile unit and the Iridium gateway; one side of the transmission always involves a mobile unit.

The Spnet2 Provisioning System, provided by Iridium, can be used to modify the delivery e-mail address for any of the SBD demo units in their account; monitor the activity and view reports for the SBD demo units in an account; send MT text messages directly to an SBD demo unit in an account; and delete MT messages that are pending in the download queue for an SBD demo unit in an account. This system was very useful in the testing process as it provided the ability to monitor the gateway queue and the ability to see the messages reach the gateway and subsequently be processed. This provisioning capability is not available to the system users.

8 Results

8.1 CSD Results

A series of experiments were performed to baseline the client-server application performance on the same machine, in the same local area network (LAN) and dial-in service. These preliminary experiments are described in [2]. These tests were run interactively, with users typing the text transmitted and received by the echo client. The packet sizes were variable, with the average packet size for all the dynamic and static experiments was 14 Bytes.

Our circuit-switched experiments used the Iridium network and land-line as the spoke into the hub. The Iridium gateway in Phoenix, AZ, was used. The Iridium satellite network carried traffic from the client laptop to the ground station in Arizona (2008 miles). The trip to the hub in Laurel was via land-line (1989 miles). They were run on both static and on dynamic platforms. The static experiments results, shown in Figure 1, had an average RTT of 1686 msec. Dynamic tests were conducted shipboard in Annapolis, MD. Dynamic experiments had an average RTT of 1812 msec. This greater average was unlikely due to the moving platforms when using Iridium; the lowest RTT was recorded during a dynamic test (981 msec). More likely factors are traffic, routing delays, or the error correction Iridium uses for data service. Figure 2 contains results of the dynamic tests.

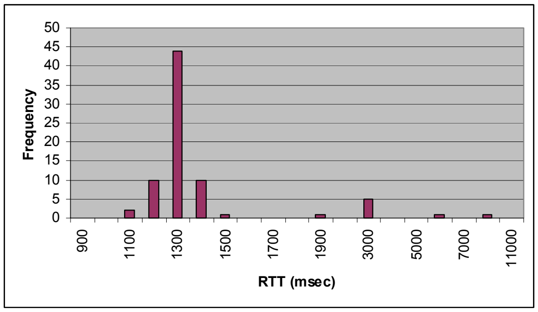

Fig. 1. Static Iridium Test RTTs

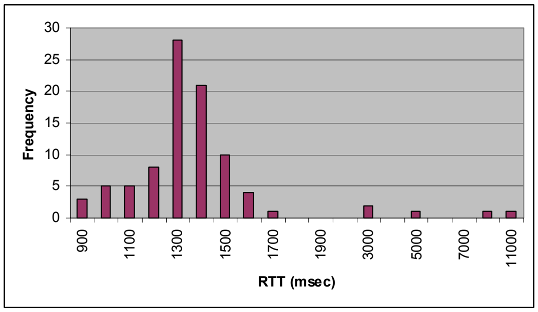

Fig. 2. Dynamic Iridium Test RTTs

The combined data in the experiments had an average RTT of 1755 msec. This combined data is shown graphically in Figure 3.

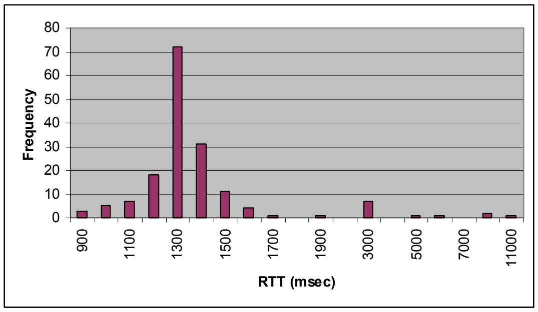

Fig. 3. Combined Iridium Test RTTs

Statistical analysis indicated that there was no correlation between packet size and round-trip time. This was as expected, since each text message was sent in one UDP datagram. Table 1 presents the summary statistics for the Iridium tests. The modes for both the static and dynamic tests are between 1300 and 1400 milliseconds, providing us a more realistic round-trip time than using the average time.

Table 1. Summary statistics for the Iridium tests

	Static	Dynamic	Combined
Average RTT (msec)	1686.21	1811.89	1755.11
Standard Dev	1199.18	2059.99	1721.50
Max RTT (msec)	8832	16073	16073
Min RTT (msec)	1161	981	981
Mode (msec)	1362	1332	1332
Average Packet Size (Bytes)	11.48	16.14	13.82
Percent <= 1700 msec	89.33	93.41	91.57

8.2 SBD Results

The information provided summarizes at a high-level the findings of the SBD investigation project using commercial Iridium services. All aspects of the SBD service were exercised thoroughly. The initial impression is that the service is still relatively new and just positioning itself for production usage. During the course of testing various problems were encountered, one of which derailed testing for three days (Iridium e-mail process was generating incorrect random e-mails to the device queue) and others that are planned to be fixed or enhanced in the next release(s) (e.g. mismatching MT message sequence number (MTMSN), socket access, MT Ring Alert, better MT status processing). Although there were issues, the service worked reasonably well, with the MO processing appearing to be the more reliable and controllable in comparison to MT processing. After completing this phase of the project, the team was exposed to the military version of the SBD service. In the initial investigation it was apparent that the military version of the service had enhanced functionality that was not available in the commercial service (e.g. direct Internet Protocol (IP) access, ftp access, MT ring alert).

Table 2 shows the average modem processing times in milliseconds running in Full Duplex mode given MO messages of size 100, 500, 1500 Bytes and MT message sizes of 100, 300, 500, 1000, and 1500 Bytes. These packets were automatically generated by the applications, rather than by interactive users. While running in Full Duplex mode, some of the MO sends had corresponding MT receives in the same modem process, while some had no MT transmissions waiting in the queue. The "Overall" average includes the modem processing time for all MO transmissions, while the "Simult MT" includes the modem processing time for only those MO transmissions that included MT transmissions. The modem processing time increased significantly when the MT transmissions were involved and increased more based on the size of the MT message. Figures 4, 5, and 6 graphically depict the values for MO sizes of 100, 500 and 1500 Bytes respectively.

Table 2. MO/MT Full Duplex Processing Times (in milliseconds)

Start of MO Send	End of Mo Send	Attempts	MT Size	Complete Avg Overall	Complete Std. Dev. Overall	Simult MT Avg Overall	Simult MT Std. Dev. Overall
MO SIZE = 100							
10/22/2004 18:00:22	10/22/2004 18:46:54	100	100	7144.99	4158.70	7531.20	3189.49
10/22/2004 20:00:29	10/22/2004 20:48:54	100	300	7771.56	4170.38	9904.02	4345.30
10/22/2004 22:00:24	10/22/2004 22:51:58	100	500	8848.29	6269.02	11021.74	4651.70
10/23/2004 0:00:20	10/23/2004 0:59:48	100	1000	10507.37	6254.22	16021.23	2871.59
10/23/2004 2:00:27	10/23/2004 3:05:13	100	1500	10462.00	7181.99	12985.82	7817.07
MO SIZE = 500							
10/23/2004 4:00:25	10/23/2004 4:51:17	100	100	9344.14	3375.93	10392.90	3905.87
10/23/2004 6:00:25	10/23/2004 6:52:32	100	300	10018.98	2976.48	10730.27	2444.94
10/23/2004 8:00:28	10/23/2004 8:56:59	100	500	11227.23	3827.96	12730.53	2383.01
10/23/2004 10:00:27	10/23/2004 10:59:05	100	1000	11726.64	5162.25	17504.46	3708.66
10/23/2004 12:00:25	10/23/2004 13:08:03	100	1500	13396.71	7646.00	22042.86	2974.75
MO SIZE = 1500							
10/23/2004 14:00:26	10/23/2004 15:06:39	100	100	18448.98	3164.96	19598.36	2917.91
10/23/2004 16:00:25	10/23/2004 17:09:22	100	300	18424.85	3989.88	19910.96	2757.66
10/23/2004 18:00:25	10/23/2004 19:10:35	100	500	19369.53	3644.47	21266.19	3743.87
10/23/2004 20:00:25	10/23/2004 21:22:45	100	1000	22199.43	4822.07	25479.39	2120.58
10/23/2004 22:00:24	10/23/2004 23:21:19	100	1500	20843.81	5613.40	26034.64	2429.00

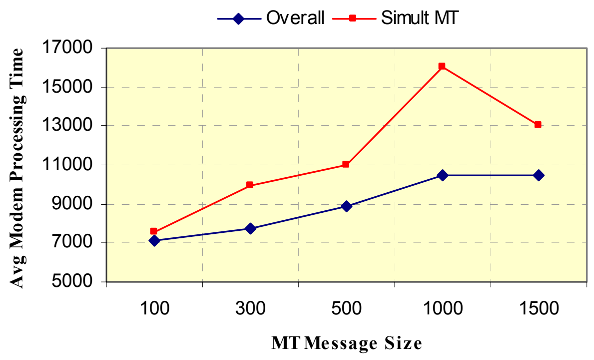

Fig. 4. MO Size 100 [Full Duplex Mode]

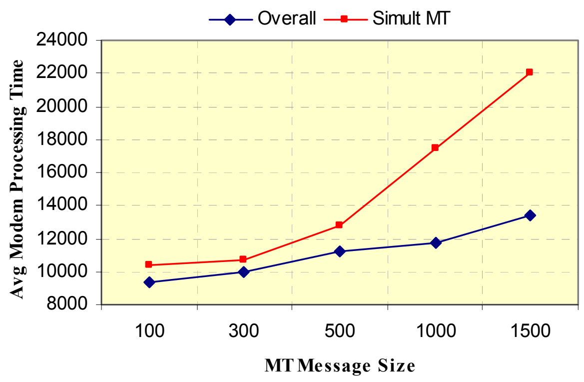

Fig. 5. MO Size 500 [Full DuplexMode]

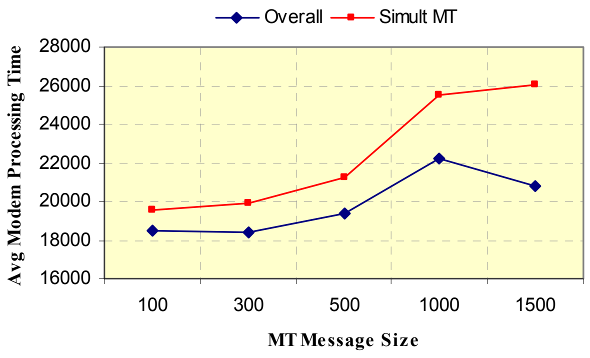

Fig. 6. MO Size = 1500 [Full Duplex Mode]

8.3 Analysis of combined results

Although the time spent setting up the call was an issue for the CSD experiments, that time was not reflected in the data sets; only actual transmission time was captured.

The SBD experiments focused on the processing time to send or receive on mobile equipment, and did not include the component of travel through the constellation.

Given that the data transfer rate for CSD was more than two times faster than that of SBD, we would normally expect the CSD latency to be half of the SBD service. However in the CSD experiments, the distance that the CSD packet had to travel via land-lines to and from the hub accounts for the longer time estimated for a packet approximating the SBD packet size.

In addition to the differences in the data rate of the two types of Iridium service, there were several other differences in the experiments: distance from the gateway, simultaneousness of incoming and outgoing transmissions, and packet size used for the experiments.

The distance from the gateway in Tempe was approximately 2,915 miles for the SBD tests, and 2008 miles for the CSD experiments. Each actual distance is increased by the twice the altitude of the constellation orbit, 500 miles, for the up- and down-link. The CSD experiments incurred additional travel time due to the use of land-lines into the hub.

CSD client and server were built as blocking applications and not capable of simultaneous transmissions. Results for the one-way and simultaneous SBD messages were captured.

The CSD applications were interactive, and as a result had variable length and shorter messages than the SBD experiments. The average packet length in the combined CSD experiments was 14 Bytes, which was approximately 1/7 of the smallest SBD messages. The processing time for the smallest MO (100 Bytes) and smallest MT (100 Bytes) value was just over 7 seconds. Without having the actual latency to the gateway, a very rough estimate can be made using the distance to the constellation, though the constellation and down to the gateway (2915 + 1000 miles). (The positions of the satellite need not be directly over the gateways.) This estimate ignores queuing and switching times at the individual satellites in the constellation. Using that estimate of signal travel time adds much less than a second to the transmission time, and can be ignored.

Comparing the 7 seconds to the CSD experiments, we can multiply the approximately 1.4 seconds of the shorter 14 Byte message by 7 to approximate a 100 Byte message, yielding 9.8 seconds. This is a round trip time, giving a one-way time of approximately 4.9 seconds. While we would expect the CSD to be twice as fast as the SBD times, we assume that the discrepancy between these values is due to the land-line component part of the CSD experiments.

8.4 Mobile Originated Insights

Through discussions with Iridium, the optimal economic message size of an MO message is approximately 300 Bytes (when transmissions are regularly greater than this, one of the other Iridium services may be better suited). In technical testing, there did not appear to be a solid basis for the optimal message size of 300 bytes. The larger the message size, the greater the throughput and it also did not appear to increase the error rate as the message size increased. If the mobile unit is in an antenna disadvantaged location, then it may be better to keep the message size down, as the connect time will be shorter, resulting in fewer errors. The average time it takes to send an MO message is between 6 and 22 seconds (processing time in the modem including signaling channel negotiations from the modem to the satellite), depending on message size. The actual end-to-end time (from start of send to receive via e-mail) basically adds 2 to 4 seconds due to the modem processing time to allow delivery of the e-mail from the gateway to the destination e-mail address. This additional time from the gateway to the destination e-mail address appears to be fairly constant across the various message sizes. Overall end-to-end times may vary by a few seconds; however these are the average end-to-end times obtained in numerous tests of the MO transmission process. Given good conditions, the average error rate on MO transmissions is approximately 2 to 3%. This error rate can increase significantly due to satellites not operating at 100% capacity or due to bad weather affecting the transmissions. When utilizing the SBD MO transmissions, there are going to be error transmissions and this should be planned for in the controlling applications.

Introducing a minimal delay of one second or more between transmissions appeared to help stabilize the process and lower the error rate. In the initial tests messages were processed sequentially as quickly as possible with no delay between sends, however, there was a lower error rate than when a delay was introduced. In a real-world production scenario, there may be delays between individual MO transmissions depending on the overarching application. Higher error rates were observed during periods where the satellite coverage in the test area was not optimum. These higher error rates should be handled by the error handling logic within the driving application.

Testing was done to evaluate transmissions during different times of the day to determine if timings and/or throughput were significantly higher during any specific period. While it did appear that transmissions ran a little faster during off hours and weekends (Hawaii Standard Time), the difference was almost negligible.

8.5 Mobile Terminated Insights

Overall the MT process was significantly different from the MO process. It was more difficult to attempt to maximize the throughput with MT transmissions as there were built-in delays at the Iridium gateway process. It was up to the receive side application to periodically check and retrieve the messages. If the sending application sends a large volume of messages, the messages will likely build up in the queue and the end-to-end (send to receive) time will increase dramatically. If the messages do build up in the queue due to a large volume of messages to process or in the case where the receive side is not functioning correctly, there is really no way to clear the messages in the queue. The receiving application processes messages one-by-one or uses a manual utility to delete the messages.

The MT message size did not have an effect on the MT Send process, however the modem processing time for the mailbox check process to retrieve an MT message from the queue averaged 6 – 20 seconds, depending on message size. There did not appear to be an optimal message size. Any optimal message size would be application and antenna location dependant. It is recommended that the sending process wait for the queue acknowledgement before sending another message. The average time from sending the message to receiving the acknowledgement was approximately 30 seconds. For all testing, if a successful acknowledgement was received from the queue, the MT message was eventually transmitted and no messages were lost. As long as all of the setup information is correct, there appears to be few instances where an error in the MT transmissions will occur. While the Iridium gateway queue may fill up, this should not be a problem if it is managed correctly by the controlling application. It appeared that attempting to correlate the message size to the physical make-up of the message, fourteen 135 byte segments and testing the boundaries provided very little benefit. A significant observation that was confirmed by Iridium is that the message sequence number (MSN) used for the MT transmissions does not match MSN on the receiving side.

Overall it appears that the SBD MT transmissions are best used for relatively short and infrequent transmissions. An application with a high volume of messages and/or large messages would not be well suited for the SBD MT transmissions. The MT transmission is not a good method to support file transfers on a regular basis. If consistent, predictable processing and end-to-end times are desired, then the controlling program should not attempt to send more than one message every thirty seconds, based on the approximate time that is required to wait for acknowledgement and allowing time for the message to be queued and retrieved by the receive side application.

8.6 Full Duplex Insights

When executing SBD in Full Duplex mode sharing the mobile unit, the application for sending MO transmissions and receiving MT transmissions must be integrated. Additionally, the application must provide the controls to optimally support simultaneous transmission and reception. Since an MO Send performs an MT Receive, when there are many more MO transmissions than MT transmission on a regular basis, then there should not be a need to perform additional mailbox. However, if the period of MT transmissions is less than the period of MO transmissions or if they are close to equal, the application should perform the mailbox checks periodically between actual MO sends.

Executing in full-duplex mode can generate a significant amount of e-mail that may be of interest to the MT sending process, the MO receiving process, or both. If the intent is to function in full-duplex, it may be more efficient to develop one e-mail client that feeds the appropriate information to the appropriate send or receive application. When running in Full Duplex mode the actual processing time for the modem increases significantly when an MO Send and MT Receive process are performed in the same transmission. It does not appear that the impact is equal to the sum of the MO Send and MT receive process, however the process of retrieving the MT message does impact the normal modem processing time for the MO. This also does not impact the end-to-end times of either the MO or MT transmissions; however, it does tie up the modem longer, so it is not available for performing the next transmission.

9 Conclusions

SBD and CSD each have there own characteristics that are suited to different applications. avoids the time penalty of establishing a call, and the cost of maintaining a call and may be preferable for applications that need to intermittently exchange smaller packets of data. The SBD service is certainly more appropriate than the CSD service for an automated client-server application. The savings in power and transmission signature are also beneficial to users of systems in the field.

Time-critical applications, or those applications needing large data transfers, may need to analyze whether the latencies incurred in SBD can be tolerated, or whether CSD may be more appropriate. For example in the TCN application sensor data is continuous from the viewpoint of a data provider. However, for a data user there may only be a requirement for periodic updates of a summary of current sensor data. The hub and spoke approach allows a mix of both the CSD and SBD applications depending on the needs of the end-user.

10 Future Work

Gathering data about the complete end-to-end latency for the SBD service will round out the experimental data already collected. Additional CSD experiments using comparable message sizes would also give a practical verification of the delays and support further comparisons between the service types. Eliminating the large land-line component in the CSD experiments, by using a hub closer to a gateway, would also be important in a more consistent comparison of the Iridium constellation for both types of service.

An examination of the military version of SDB and the enhanced features of the service, over what is now available commercially, should yield valuable insights that may not affect performance directly, but may better support users in the field.

Acknowledgements

Raytheon Solipsys supported both research projects. For the CSD experiments they loaned the equipment used in the experiments. Many thanks go to LCDR Lori DeLooze, USN, for her invaluable participation in CSD testing. Much gratitude goes to the students of the Spring 2004 SI455 Advanced Networks of the US Naval Academy course for their assistance in carrying out the shipboard portion of the experiments.

References

[1] Lemme, P.W., Glenister, S.M., and Miller, A.W., "Iridium(R) Aeronautical Satellite Communications", Aerospace and Electronic Systems Magazine, IEEE , vol. 14 no. 11, 1999, pp 11 –16

[2] McMahon, M. M. and Firkin, E. C., "Performance of a Hub-based Network-Centric Application over the Iridium Satellite Network", accepted by the 4th International Conference on Networking (ICN 2005), Reunion Island, April 17-21, 2005.

[3] NexGen Communications LLC, "An Introduction to Short Burst Data (SBD)", prepared for Enhanced Mobile Satellite System (EMSS) Program, 5 April 2004, Available from https://sbd.pac.disa.mil

[4] Personal Satellite Network, Inc., "Iridium™ SMS and SBD FAQ", Jan 2005, available from http://www.skyhelp.net/acrobat/jan_05/Iridium%20SBD-FAQ%201-05.pdf

[5] Pratt, S.R., Richard A. Raines. R.A., Fossa JR, C.E., and Temple, M. A., "An Operational and Performance Overview of the Iridium Low Earth Orbit Satellite System", IEEE Communications Surveys, Second Quarter 1999 vol. 2, no. 2, 1999available from http://www.comsoc.org/livepubs/surveys/index.html