Real-Time Virtual Instruments for Remote Sensor Monitoring Using Low Bandwidth Wireless Networks

Biruk A. Gebre, Liwen Guo, Nishit Patel, Kishore Pochiraju
Design and Manufacturing Institute, Stevens Institute of Technology
Hoboken, New Jersey 07030, United States

ABSTRACT

The development of a peer-to-peer virtual instrumentation system for remote acquisition, analysis and transmission of data on low bandwidth networks is described. The objective of this system is to collect high frequency/high bandwidth data from multiple sensors placed at remote locations and adaptively adjust the resolution of this data so that it can be transmitted on bandwidth limited networks to a central monitoring and command center. This is achieved by adaptively re-sampling (decimating) the data from the sensors at the remote location before transmission. The decimation is adjusted to the available bandwidth of the communications network which is characterized in real-time. As a result, the system allows users at the remote command center to view high bandwidth data (at a lower resolution) with user-aware and minimized latency.

This technique is applied to an eight hydrophone data acquisition system that requires a 25.6 Mbps connection for the transmission of the full data set using a wireless connection with 1 – 3.5 Mbps variable bandwidth. This technique can be used for applications that require monitoring of high bandwidth data from remote sensors in research and education fields such as remote scientific instruments and visually driven control applications.

Keywords
Data Acquisition, Sample Rate, Latency, Resolution, Throughput, Network Bandwidth and Transmission Control Protocol (TCP)

1. INTRODUCTION

Wireless networks are a very attractive method of data transmission for remote data acquisition applications. In addition to their obvious advantages of eliminating the need for costly wired network infrastructures and being more convenient, wireless networks also enable data transmission from harsh environments and locations, which may be difficult or impossible to access by wired networks. Wireless networks also offer much greater freedom in terms of mobility and area coverage [1].

For all their advantages however, wireless networks have a few major limitations. One of the main limitations of wireless networks is their speed as, unlike their wired counterparts, wireless networks have a highly constrained and variable link bandwidth [2 - 4]. While this may not be a big concern for simple low bandwidth applications it is a major problem for high bandwidth real-time applications, which require a guaranteed high network bandwidth in order to function properly. Other limitations of wireless networks include their susceptibility to losses due to various kinds of interference, their reduction in throughput performance due to attenuation of radio signals with distance and performance degradation due to attenuating objects in the radio signal path [3, 4].

The use of the ubiquitous Transmission Control Protocol (TCP), which is not optimized for use in wireless networks, is another factor that can reduce the available bandwidth of wireless networks [5 - 8]. Thus applications using TCP on a wireless network may experience lower and more time varying bandwidth availability and should therefore adapt their transmission throughput rates accordingly.

In this paper a method to continuously transmit high bandwidth, real-time data, on a TCP wireless network is described for a sample remote data acquisition application. The described system enables the operator at the central monitoring and command center (CMCC) to dynamically adjust the throughput of the system to match that of the wireless network being used, so that data latency is minimized. The system allows the operator to visualize the instantaneous characteristic of the link and, if necessary, make a tradeoff between the latency and resolution of the data to help maintain the real-time nature of the system and better utilize the available network resources. Automated Controls strategies have also been implemented into the system to enable dynamics adjustments of the system throughput to minimize latency while maximizing data resolution.

2. METHODOLOGY

The system consisted of two main components (Figure 1). The first component, the remote data acquisition component, is responsible for the acquisition, analysis, reduction and transmission of data collected from the sensors. In addition this component is also responsible for executing any commands that are transmitted to it from the second component. The second component, the local visualization and command component, is responsible for displaying the data it receives from the remote component, executing algorithms for throughput and latency control, as well as sending commands back to the remote component. The two components communicate with each other over a wireless TCP/IP network.

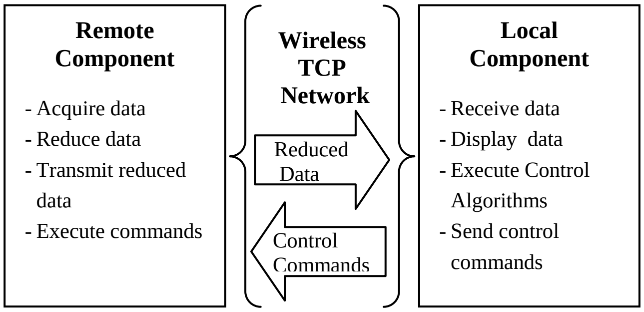


Figure 1. Basic system architecture

A brief description of both of the Remote and Local components is given in the following sections.

Remote Component

The main functions of the remote component (RC), as illustrated in Figure 2, include acquiring raw data from the sensors, carrying out any needed analysis on the data, reducing the data to a smaller sample / bit rate, and then transmitting the data through a constrained wireless network connection.

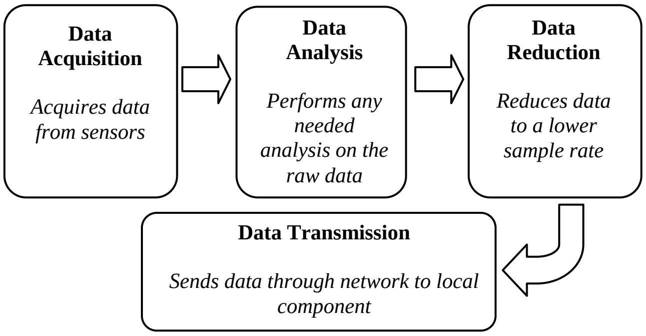


Figure 2. Main functions of the remote component

Acquiring raw data from the sensors is the first main function that the RC performs, as this data is needed to carry out the remaining functions of the remote component. In order to do this, sensors need to be connected to a data acquisition system. The data acquisition system reads the raw data from these sensors, at a constant rate, and passes this data along to the data analysis function.

The analysis done on the raw data is dependent on the sensors being used as well as the information that needs to be extracted from the data collected. Microphones, for example, pick up varying acoustic pressure waves and output analogously varying voltage signals. If the user is interested in viewing the frequency characteristics from the voltage signals then the data analysis portion of the system would carry out spectral analysis on the voltage data and output the spectral information of the acoustic signals picked up by the sensors. Other sensors may have outputs that are proportional to the signal they measure. In this case the data analysis involved would be a simple conversion of the data to the appropriate unit.

Once the proper analysis has been carried out and the appropriate data information has been obtained, the processed data is sent to the data reduction function which will reduce the data to a smaller bit rate. The reduction of the data is performed by reducing the number of data points that are being transmitted. A number of different data reduction methods can be used. These reduction methods included averaging over a group of data points, choosing singular data points that meet a specific criterion from a group of data points, or just simply reducing the sample rate of the data by only transmitting a subset of the data. Each one of these reduction methods work by reducing the resolution of the data, therefore a tradeoff will exist between the bit rate of the data and the resolution of the data.

The final function of the remote component is to transmit the reduced data to the local component. In order to ensure that no data is lost during transmission and that all the data is received in the proper order a network protocol such as TCP, which guarantees reliable and in-order delivery of data, is used. If any other information is also being transmitted along with the data then the transmitted data is encoded in a simple message format that is decoded when the message is received by the local component.

Local Component

The local component (LC) is located at the CMCC and has four main functions (Figure 3). The first is to receive and, if necessary, decode the data sent by the remote component. This involves establishing a TCP/IP network connection with the remote component through which the data is received, and then decoding the received data using the proper messaging format. Once the data has been received and decoded, the second function of the local component is to display this data using the appropriate plots.

The LC is also responsible for defining the resolution and thus the throughput of the data that is transmitted by the RC. The resolution of the data can be defined either by the user at the CMCC or by automated control algorithms. Statistics such as the data latency, throughput of reduced data and the network bandwidth being utilized are displayed and plotted at the LC to aid the user in defining the resolution of the data. The last function of the LC is to send commands, such as what the resolution of the reduced data should be, back to the RC. These commands allow the user at the CMCC to control the various functions of the remote component.

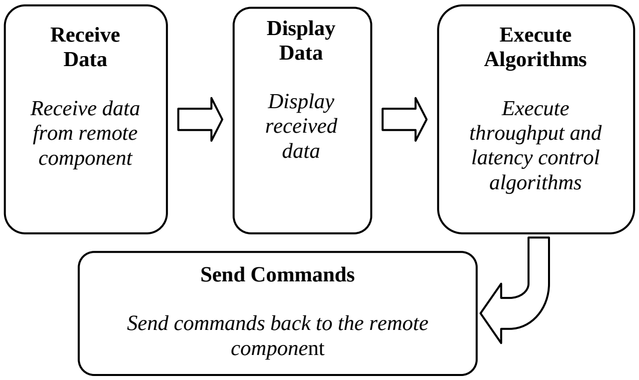


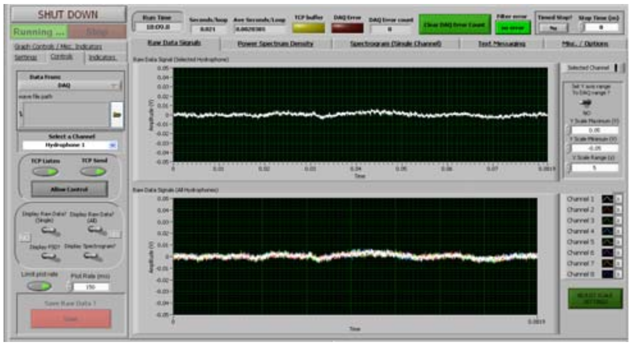
Figure 3. Main functions of the local component

3. ILLUSTRATIVE EXAMPLE

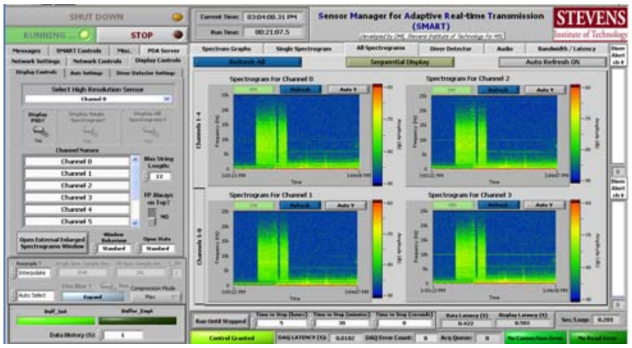
To help better explain the methodology, an actual application in which this system is being used for a Maritime Systems

Laboratory research project is described in this section. For this specific application the system is used to transmit spectral data from eight hydrophones situated at a remote offshore location. The hydrophones are used to pick up acoustic signals underwater. Similar to microphones they work as sound to electricity transducers. The hydrophones are placed a couple of hundred feet offshore and a wireless network is used to transmit the data from the remote component of the system to the local component.

In this application both the remote and local components of the system were programmed as Virtual Instruments (VI's) using LabVIEW, a graphics programming language. Using LabVIEW, graphical user interfaces (GUI's) were created to allow the users to interact with the Virtual Instruments (Figure 4).



Remote Component Front Panel



Local Component Front Panel

Figure 4. Graphical user Interfaces for Remote and Local VI's

Remote Component
The Remote VI performs all the main function for the remote component. A description of the remote VI as well as the setup of the remote component is given in the following sections.

Data Acquisition: To obtain the voltage data from the hydrophones, each hydrophone is connected to a custom-made 68-pin hydrophone terminal. This terminal outputs the hydrophone voltage reading from eight different input jacks to a 68-pin connector. A 68-pin ribbon cable is used to connect the terminal to a National Instruments PCI – 6123 data acquisition (DAQ) card [9] that is mounted in a free PCI slot on the computer running the Remote VI. This DAQ card has the ability to acquire data from eight different channels simultaneously at a sample rate up to 500 Kilo-samples per second (KS/s) per channel. For our purpose the Remote VI is programmed to acquire data from eight channels (one for each hydrophone) at a sample rate of 200 KS/s per channel. Once the user runs the VI, the DAQ card acquires the voltage reading from the eight hydrophones at the specified sample rate and outputs it to the VI.

The data acquisition system described above is placed on a small boat at an offshore location somewhere close to where the hydrophones are deployed. Figure 5 illustrates how this data acquisition system is setup.

.

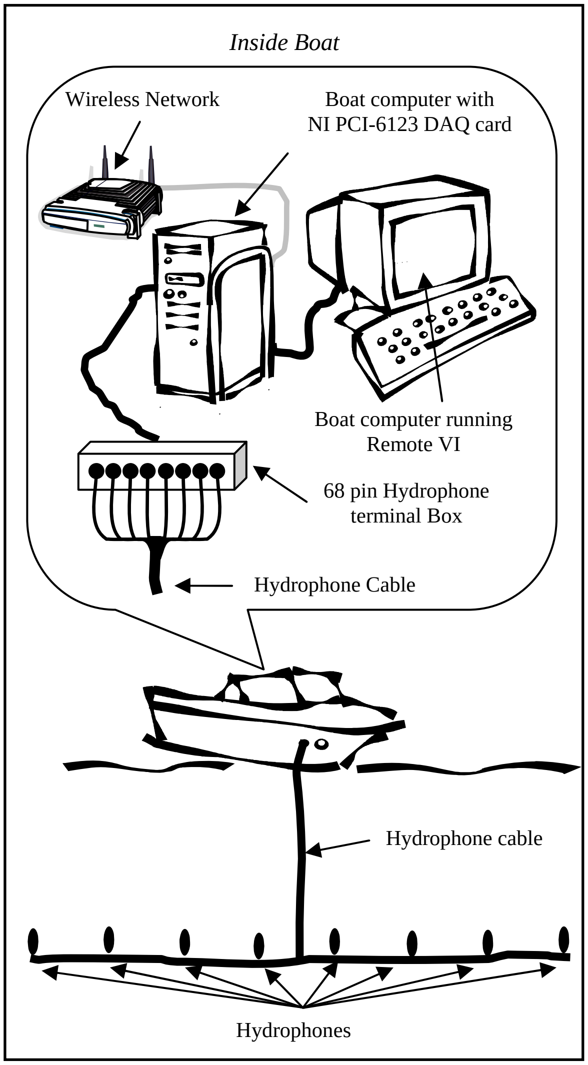


Figure 5. Setup for Remote component

Data Analysis: The main information of interest is the frequency characteristics of the acoustic signals detected by the hydrophones. Therefore after the voltage data is acquired by the DAQ card, a fast Fourier transform (FFT) spectrum analyzer function is used to calculate the power spectral density of the voltage signals. The output from this spectral analysis function is an array of spectral magnitude measurements of 100 KS/s for each hydrophone in the frequency range of 1Hz – 100 KHz.

In addition to specifying the sample rate the user running the VI also has the ability to choose how many samples to read at a time from the data acquisition function and analyze using the

FFT spectrum analyzer. The number of samples the user reads at a time determines the resolution of the spectral measurements. The resolution of the spectral measurement (S_{res}) is equal to the sample rate (SR) divided by the number of samples (N_s) acquired (S_{res} = SR/ N_s). For example, if the user decides to read 200 KS at a time then the data acquisition function will output 200 KS of voltage readings from each hydrophone every second. After the VI performs spectral analysis on this data the result will be spectral measurements of 100 KS [11]. Since the program calculates the spectral readings for a frequency range of 1Hz – 100 KHz, this corresponds to a spectral resolution of 1 Hz per sample. On the other hand, if the user decided to read data 20 KS at a time then the data acquisition function will output 20 KS of voltage readings every 0.1 seconds and this will result in spectral measurements of 10 KS. Again since the frequency range is 1 Hz – 100 KHz this will result in a resolution of the spectral measurements of 10 Hz per sample. Although the user can change the number of samples to read at a time, the sample rate for both the data acquisition function (200 KS/s) and the spectrum analysis function (100 KS/s) always stay the same.

The spectral measurements are placed in an eight-column array (one column for each channel). This array is then queued inside a data buffer in the order it is received. Both the data acquisition and the data analysis functions have been placed in the same while loop, therefore the program will continuously keep acquiring data, performing spectral analysis, and inputting data in the buffer at the constant rate specified by the user until the remote VI is stopped.

Data Reduction: For this specific application a wireless network is used that has a measured available bandwidth of about 1 – 3.5 Mbps. Transmitting the raw data from the sensors (1.6 MS/s or 25.6 Mbps) or the spectral data at a sample rate of 100 KS/s per hydrophone (800 KS/s or 12.8 Mbps for all eight hydrophones) is therefore not feasible as the network will not be able to transmit data at such a high bit rate. Reduction of the data to a smaller sample rate is therefore needed. In order to do this a sub-VI was written to reduce the data using one of three different reduction modes named Mean Reduction, Max Reduction, and Min Reduction.

All three data reduction methods work by reducing the resolution of the spectral measurements; they take a specified number of spectral measurements and reduce them into a single spectral measurement. So for example, to convert 100 KS of spectral measurements (1 spectral reading for every Hz) into 10 KS of spectral measurements (which equates to converting 100KS/s into 10KS/s), the data reduction sub-VI takes every bin of 10 spectral measurements (1Hz – 10Hz, 11Hz – 20 Hz, 21Hz – 30 Hz …999,991 Hz – 100KHz) and converts them into a single spectral measurement which represents the corresponding frequency range (Figure 6). Doing this reduces both the number of samples and the resolution of the spectral measurements. How the sub-VI converts the bin of spectral measurements into a single spectral measurement depends on the reduction mode used.

.

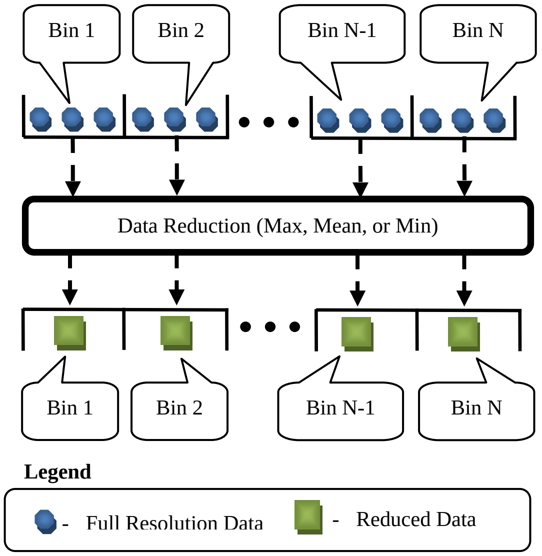
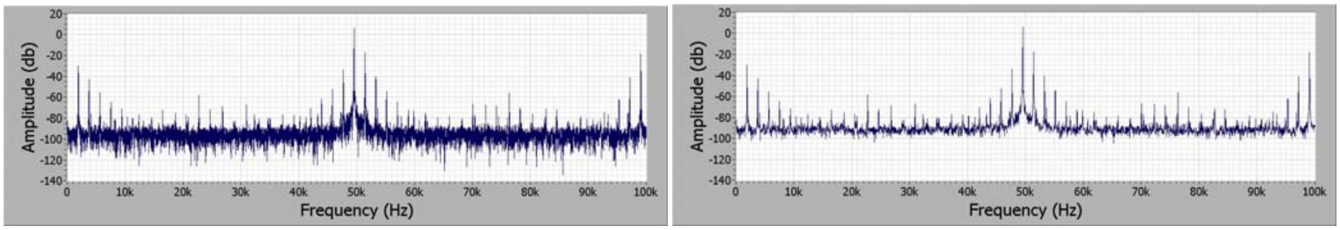


Figure 6. Data Reduction

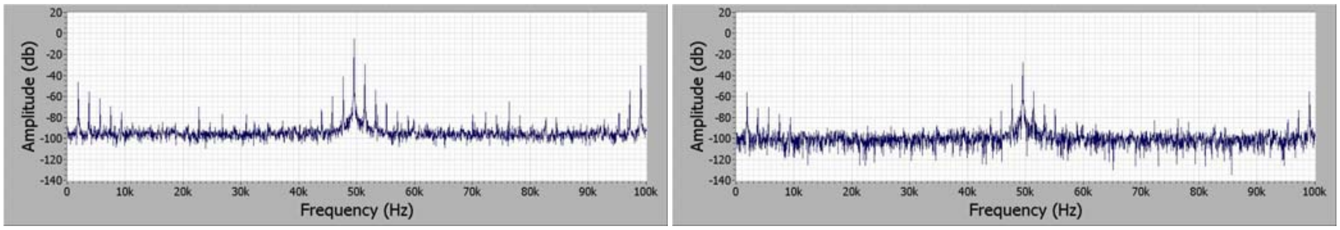
In the Mean Reduction mode the sub-VI calculates the mean value of the bin of spectral measurements and assigns this mean value as the single spectral measurement for the frequency range of that bin. The Max Reduction method picks the maximum value from the spectral measurements and assigns this as the single spectral measurement for the frequency range. Similarly the Min Reduction mode picks the minimum spectral measurement value and assigns this as the single spectral measurement for the corresponding frequency range. Figure 7 shows the plots of a spectral graph that has been reduced from 10 KS to 2.5 KS using the three different reduction modes. From the plots it can be seen that the Max Reduction mode is good for preserving most of the high peaks in the signal after Reduction, while the Min Reduction mode is good for preserving the low peaks. The Mean Reduction mode provides a mix of both the max and the min reduction mode and produces a result in between the two extremes.

For the majority of the time the information of interest in most frequency plots are the high peaks, therefore the Max Reduction mode would be the most suitable reduction method to use. To demonstrate how the signal quality is affected by the resolution, Figure 8 shows plots of a spectral graph that has been reduced to three different resolutions using the Max Reduction method along with the transmission bit rates associated with transmitting data for eight channels using these resolutions. From this figure the degradation of the signal frequency resolution is evident, however it can be seen that, even at 5% of the original resolution, most of the peaks from the original signal are still present. Another important thing to note is that the reduction in data throughput is proportional to the reduction in resolution. This means that by transmitting at 5% of the full resolution, the data throughput can also be reduced to 5% of the throughput at full resolution.



Original Spectral graph at 10KS

Spectral graph reduced to 2.5 KS (Max mode)



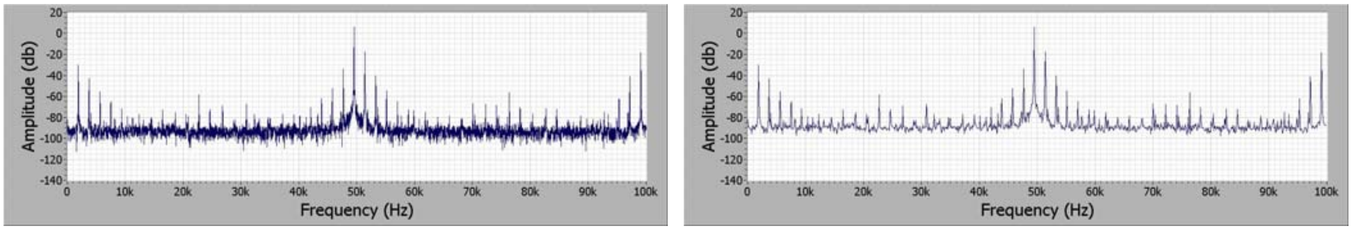
Spectral graph reduced to 2.5 KS (Mean mode)

Spectral graph reduced to 2.5 KS (Min mode)

Figure 7. Plots of the three different reduction methods

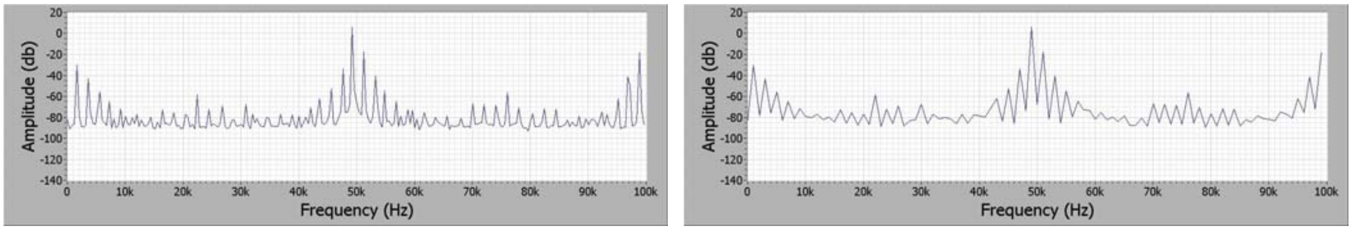
The data reduction sub-VI is located in a parallel while loop to the data acquisition and data analysis function. It reads the original spectral data placed in the buffer by the data analysis function and reduces it to the sample rate specified by the user.

It should be noted that more efficient and even lossless compression methods could be employed to improve the signal quality of the compressed data; however they are not used in this system due to the real-time constraint imposed on the system. While the lossy compression methods described above do result in some degradation of the original signal during compression, they are relatively fast at data reduction making them more suitable for real-time transmission applications. In addition, these methods use less computational resources compared to other more complex reduction methods, thereby freeing up these resources for more computation intensive processes, such as other signal processing, and user interface and display processes. The flow of data in the Remote VI is shown in Figure 9.



Original Spectral graph at 5 KS resolution (**Bit rate: 12.8 Mbps**)

Spectral graph reduced to 1 KS resolution (**Bit rate: 2.6 Mbps**)



Spectral graph reduced to 250 S resolution (**Bit rate: 0.64 Mbps**)

Spectral graph reduced to 100 S resolution (**Bit rate: 0.26 Mbps**)

Figure 8. Plots of three different resolutions using the max reduction method

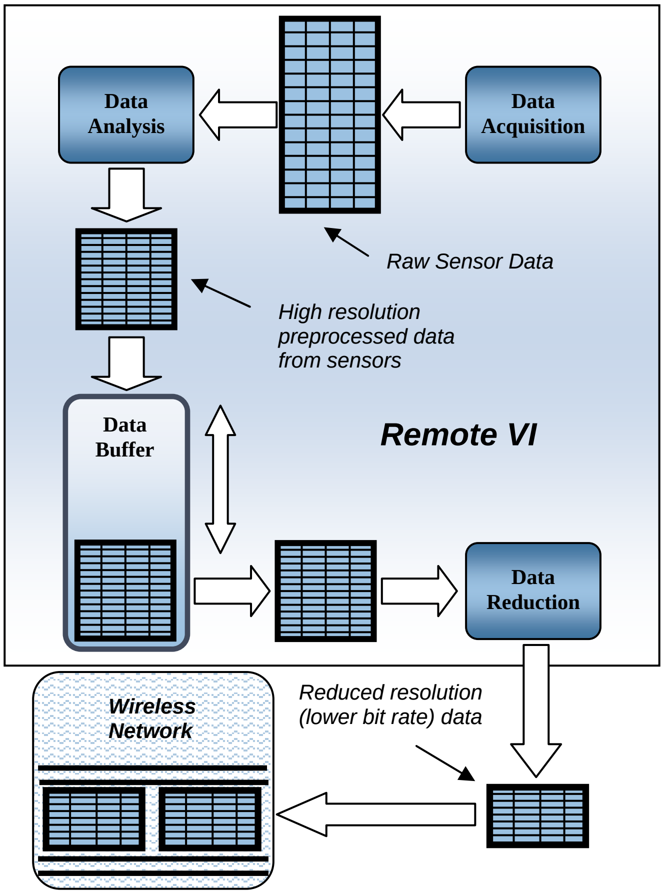


Figure 9. Flow of Data in Remote VI

Data Transmission: While the Remote VI acquires, analyzes, and reduces data, it simultaneously listens for a TCP connection on the wireless network. Once a TCP network connection is established by the Local VI, the Remote VI automatically sends the reduced data through the wireless network to the Local VI.

Due to the movement of the boat, as a result of waves and wakes from other boats, as well as possible interference from other radio signals, the bit rate of the wireless signal constantly fluctuates. In rare cases the wireless signal could even temporarily lose connection. During these conditions when the wireless bit rate is low or when the wireless connection is lost, the amount of data stored in the buffer increases as data is inserted into the buffer faster than it is extracted. When the bit rate for the wireless connection increases the buffer will begin to empty out as data is extracted from the buffer faster than it is added. For this to work, however, the bit rate of the reduced data needs to be much lower than the available bandwidth of the wireless network. The advantage of using a buffer is that it enables the system to be more resistant to data lose during fluctuations in network bandwidth, the downside of using a buffer is that it can introduce data latency into the system during cases when the wireless signal has low transmission speed.

Although the main purpose of the Remote VI is to transmit data to the local monitoring and command center, it can also be used as a standalone application to acquire and analyze the data generated from the hydrophones. Users can use the front panel of the Remote VI to view plots of hydrophone data as well as control and view how the VI is functioning using the various controls and indicators located on the front panel. In addition, this VI also has the ability to save the raw voltage data generated by the hydrophones and communicate with users if the Local VI using text messages.

Local Component

As mentioned previously another VI, named Local VI, was written to receive and display data at the local command and monitoring center. As soon as this VI is run by the user at the CMCC it automatically opens a TCP network connection with the remote VI and starts receiving the reduced spectral data. The data being sent by the remote VI is encoded in a simple messaging format, as information about the data and the remote VI is also being transmitted along with the actual reduced spectral data. The same messaging format is therefore used by the local VI to decode the transmitted data. Once the reduced spectral data is extracted it is displayed on graphs similar to those found on the remote VI.

At the same time the Local VI is receiving data it is also sending commands from the user at the CMCC back to the remote VI on the boat. The commands that are sent back to the Remote VI include the desired resolution of the reduced data, the reduction mode to use, the limit of the buffer and whether or not the VI should save the raw voltage data from the hydrophones. As soon as the Local VI establishes a connection with the Remote VI it will take over control of all the essential functions of the remote VI. User on the boat will still be able to view the data from the hydrophones as well as control nonessential functions of the VI that do not affect the transmission of data to the local VI. The users of the remote VI, however, do have the ability to take over control of the remote VI and only allow monitoring privileges to the Local VI.

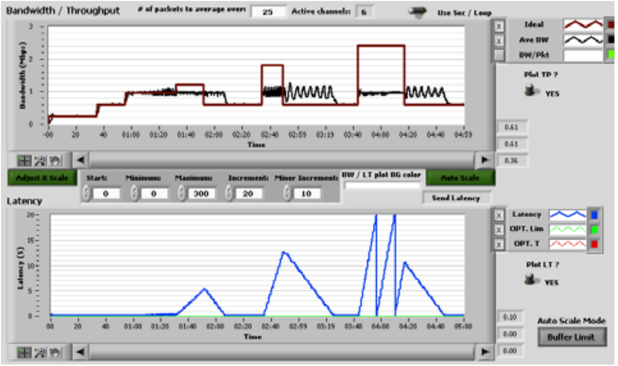
The User Interface of the Local VI is similar to the remote VI with the exception of a few differences. Due to the fact that only spectral data is being transmitted, the local VI does not have a graph to display the voltage data from the hydrophones. Instead it has three graphs to display the reduced spectral data, similar to those on the Remote VI, and an additional graph to display the throughput (bit rate at which the reduced spectral data is being read in) and the latency of the data received.

Control Strategies

The introduction of the data buffer to the system introduces the possibility of increasing the data latency in the system. Since the system is being used for a Monitoring Command and Control application insuring the real-time or near real-time nature of the system is an essential aspect of the system. Therefore the amount of data stored in the buffer needs to be minimized. In order to better meet this real-time requirement of the system, three different latency control strategies were implemented. The first strategy is a completely user controlled strategy, the second is a limit or binary control strategy and the last is a real-time adaptive optimization control strategy.

Manual Control: In the first strategy, the user at the CMCC has complete control over the system and is therefore responsible for minimizing the data latency in the system. To do this the user manually adjusts the resolution / throughput of the data to be less than or equal to the current network bandwidth thereby minimizing the buildup of data in the buffer.

The current statistics of the data, such as the data throughput, the measured network bandwidth being utilized as well as the data latency in the system, are plotted in real-time on the front panel of the Local VI and can be used by the user to help determine what resolution / throughput the user should used. Figure 10 shows an example of what these plots look like. On the top plot the data throughput and the network bandwidth being utilized are shown overlaid over each other, while on the bottom plot the data latency is shown. If for example the throughput was set at a level higher than the available network bandwidth, the user would be able to detect this from the rise in latency on the bottom plot and the difference between the graphs on the top plot and reduce the throughput accordingly.



Bandwidth / Throughput

# of packets to average over: 25

Active channels: 6

Start: 0

Minimum: 0

Maximum: 300

Increment: 20

Minor Increment: 10

Figure 10. Example of plots showing data statistics

In case the user fails to adjust the throughput properly, a limit is also set on how much data can be stored in the buffer. If the buffer reaches this limit then all the data in the buffer is automatically emptied and the most recent data is transmitted instead. This allows the user to specify the maximum amount of data latency in the system, as well as limit the amount of memory used by the system. This, however, also means that all the data that was in the buffer before it was emptied will be lost.

While the manual control strategy helps the user control the resolution of the data to minimize the latency, it detracts the user from focusing on the data as the user is also preoccupied with controlling the throughput of the system. In the following strategies, the process of controlling the system throughput has been automated using control algorithms relieving the burden of throughput control from the user.

Limit Control: In the limit or binary control strategy a latency limit is used to control the throughput of the system. In this control method the user specifies a low throughput for the transmitted data that is known to be well below the available network bandwidth and defines a maximum latency limit for the system. The user can then run the VI with a desired throughput for the system. The system will try to transmit the data at the user specified throughput. However, if at any time the data latency in the system exceeds the latency limit, the limit control automatically takes over and switches the throughput of the system to the low throughput specified by the user to allow the latency to return to the minimum value. Once the latency has been minimized the program switches back to the user defined throughput and relinquishes its control back to the user. Using this method there is no loss in data as the buffer is not allowed to fill up and the maximum latency in the system is also limited to a user defined value, thereby maintaining the real-time / near real-time nature of the system.

Figure 11 shows the response of the system while using the limit control strategy. For this figure the system was run in a controlled LAN network environment where the actual bandwidth of the network was set using external software [10]. By running the system in a control network environment the response of the system under different network conditions can be obtained.

The top plot in Figure 11 shows the system throughput (solid black line) and the measured network bandwidth averaged over 10 seconds (dotted blue line), while the bottom plot shows the data latency (solid black line), the latency limit (dotted red line) and the minimum latency (dashed blue line). For this particular figure the network bandwidth was adjusted from 5 Mbps to 4 Mbps to 3 mbps at 180 second increments. The low throughput for the system was set to 2 Mbps, the desired throughput of the system was set to 6 Mbps, and the latency limit was set to 3 seconds.

It can be seen from the bottom plot that for the majority of the time the system was able to keep the latency in the system below the latency limit. Those cases where the latency overshoots the latency limits were caused due to network delays in the commands being sent to the remote components. As the network bandwidth was decreased further and further these overshoots became more pronounced. The top plot shows that the measured network bandwidth was very similar to the values set by the bandwidth controller software, which implied that the system was fully utilizing the available bandwidth of the system. Another lobservation from the figure is that the time spent at the low throughput level increases as the network bandwidth decreases. This is of course due to the fact that as the bandwidth decreases data is extracted from the buffer at a slower rate and thus it takes longer to empty the buffer.

The response characteristics of the limit control method are important to note because when the system is run on an uncontrolled or public network, the bandwidth of the network will be unknown for the most part. In this case these system characteristics will be helpful in making a good estimation of the network bandwidth so that adjustments can be made to the desired system throughput accordingly. While the desired resolution throughput has been kept at a constant value of 6 Mbps for the test, it can be changed by the user while the system is running. Since the plots in Figure 11 are available to the operator in real-time, it allows the operator to try and match the desired system throughput to the available network bandwidth by using the information from these plots.

The best use of the limit control strategy is not for optimizing throughput. Rather the limit control is more useful for insuring that the latency limit is not exceeded and data is not lost due to buffer overflow during low network bandwidth situations and momentary interruptions in network bandwidth. As such, the desired throughput level initially defined by the user should be set to a level below the available network bandwidth for the majority of the time.

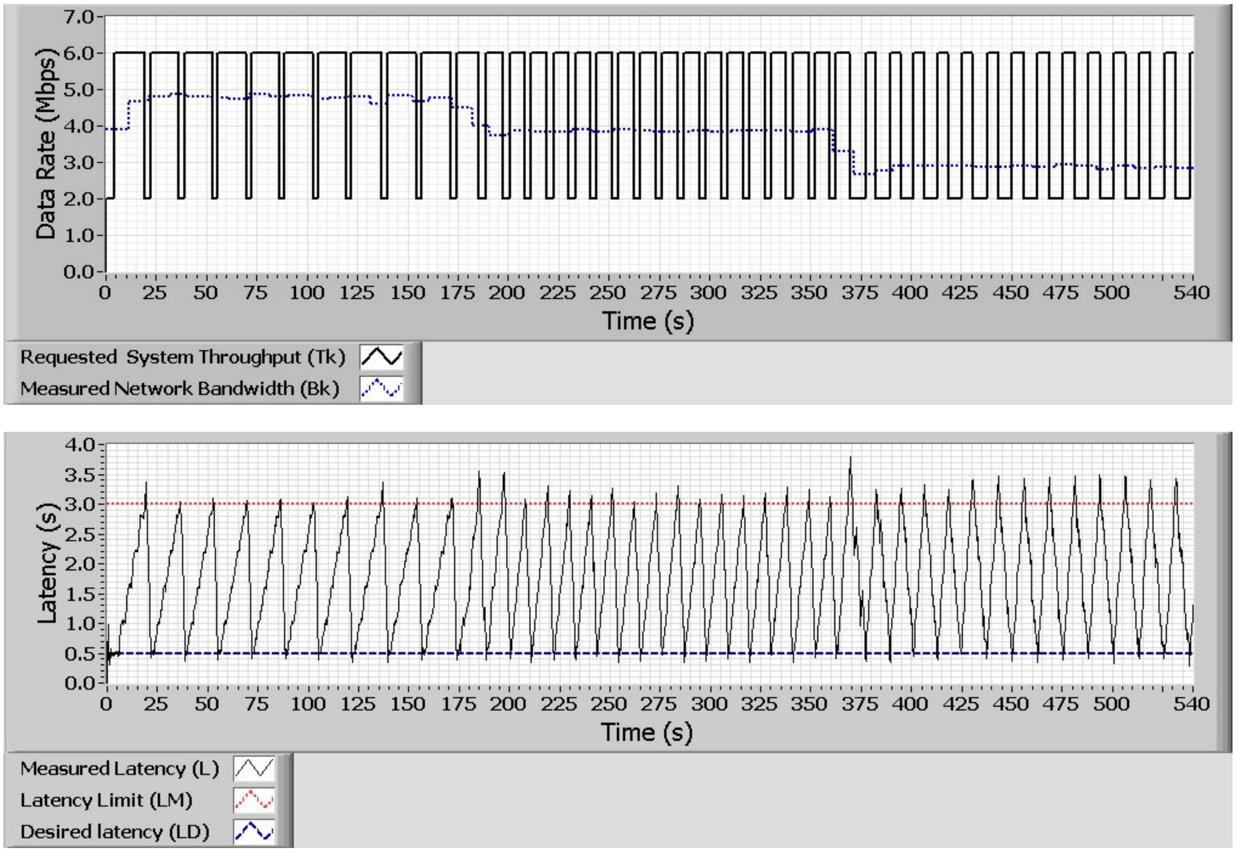


Figure 11. System response plots for Limit control strategy - network bandwidth adjusted from 5 to 4 to 3 Mbps at 180 second increments (Top plot: Requested System throughput and network bandwidth; Bottom plot: Data latency)

Adaptive Optimization Control: The last control strategy, adaptive optimization control, is a fully automated control strategy that does not require any interaction from the user once set. The strategy is used for minimizing the latency while at the same time optimizing the throughput / resolution of the system. Similar to the limit control strategy, the adaptive optimization scheme uses the resolution / throughput as the control variable and the latency as the feedback parameter. In this control approach the user specifies a target and limit latency for the system. This control strategy uses these two variables to adjust the resolution of the transmitted data so that the latency in the system matches the target latency set by the user. Once the target latency has been reached, the program will try to optimize the throughput of the system. It does this by gradually increasing the throughput of the system while using the latency as its feedback parameter to detect when the network bandwidth has been exceeded. When the latency begins to increase the system automatically reduces the resolution so that the latency is again minimized and the throughput of the system is matched closely to the network bandwidth thus optimizing the throughput of the system.

When increasing the throughput one of two different rates, a fast rate and a slow rate, is use by the control algorithm. The fast rate is used when the system is searching for the network bandwidth while the slow rate is used when the system calculates that it is in the neighborhood of the available network bandwidth. This is done so that the system can quickly find the network bandwidth while in searching mode as well as to finely adjust the system throughput when it is close to the network bandwidth.

The response of the system run using the adaptive optimization control strategy is shown in Figure 12. Similar to the limit control this test was also run using a controlled LAN network environment. For this test the network bandwidth was adjusted from 3 to 7 to 4 to 1 to 2 to 5 Mbps at 300-second increments. A stricter latency limit of 1 Second was also used.

As can be seen from the plots, the control strategy does well in finding the current network bandwidth and optimizing the system throughput while maintaining the data latency below the latency limit. In fact, the system throughput matched the network bandwidth so well that the two graphs are barely indistinguishable on the top plot. There are, however, two distinct places, the first around 360 seconds and the second at around 550 seconds, where the latency limit was exceeded. Loacating these two points on the top plot we can see that both of these cases were a result of a rapid decrease in network bandwidth. While the latency limit was exceeded the system was able to quickly recover and return the latency below the latency limit.

In Figure 13 example plots of the resolution at which the user at the CMCC would observe the transmitted data is shown for the point marked on the top plot of figure 12.

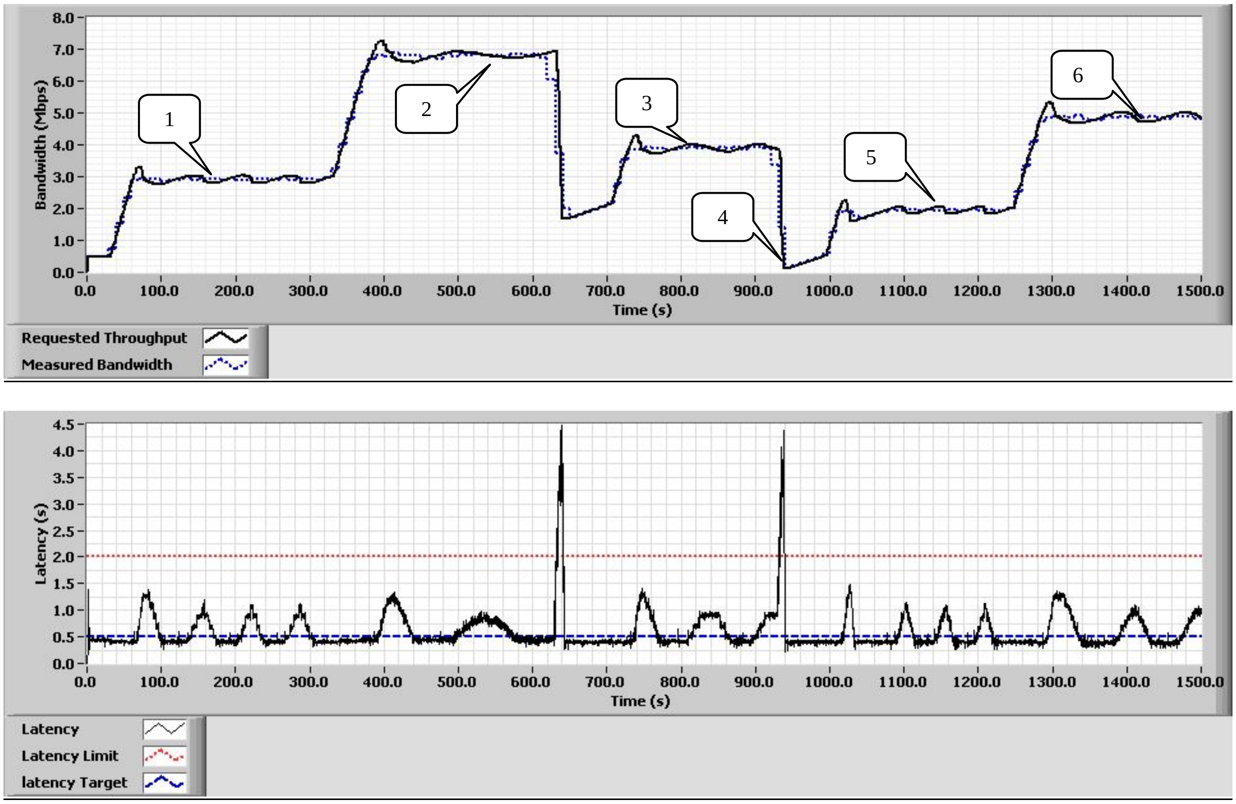


Figure 12. Controlled wired bandwidth test – 3, 7, 4, 1, 2, 5 Mbps (Top plot: Requested System throughput and network bandwidth; Bottom plot: Data latency)

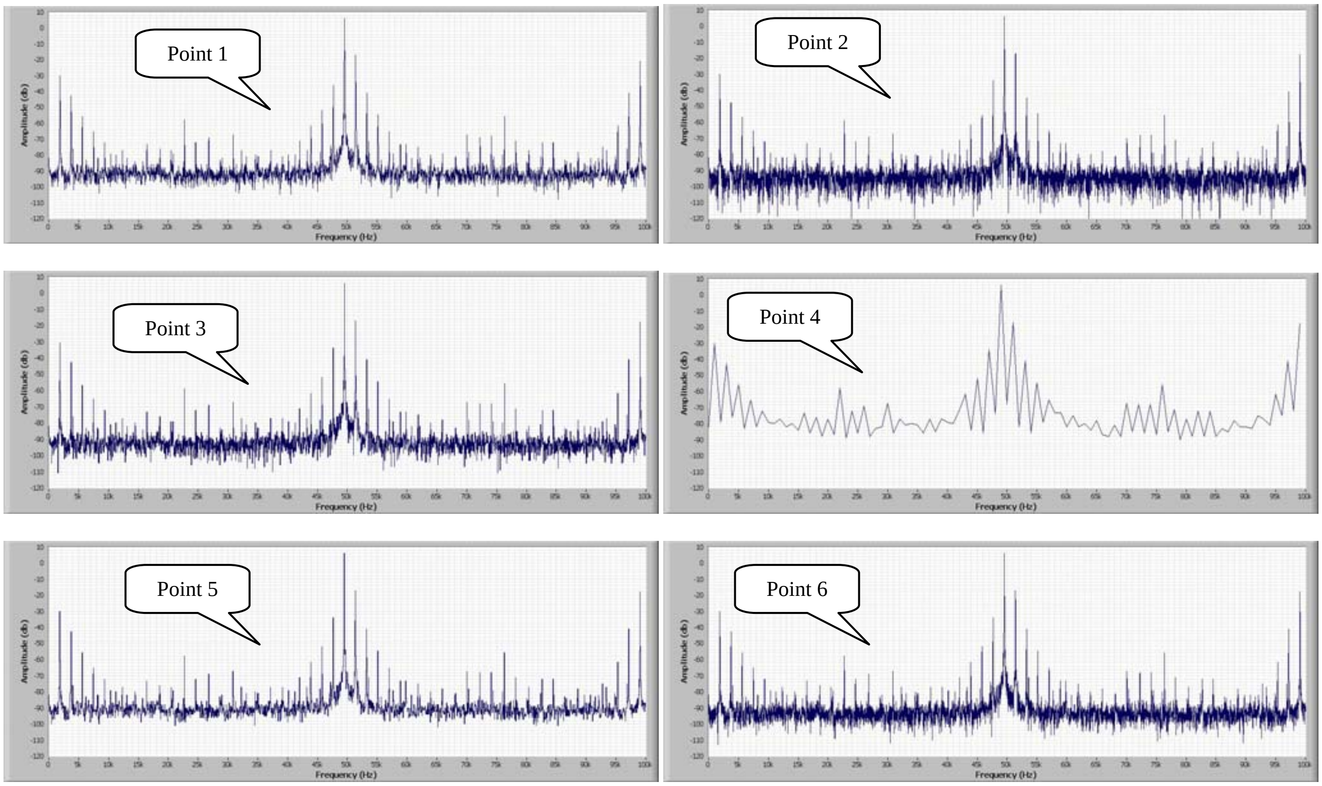


Figure 13. Example of signal resolution at different points shown in figure 12

The adaptive control strategy was also run on an uncontrolled wireless network and the results from this test are shown on the plots in Figure 14. In this test the system throughput was initially set to 0.5 mbps and the adaptive optimization strategy was enabled. This test was not run on the same wireless network used for the boat setup described above. The wireless network used was instead a public wireless network with an undetermined bandwidth which is susceptible to many of the same problems experienced by the wireless network setup on the boat. As such, the results from this test are comparable to those of the test run using the boat wireless network, and are generally typical of most public unlicensed wireless networks.

Due to the highly variable nature of the wireless network the system was not able to settle at a specific throughput for long, however it was still able to match the network bandwidth relatively closely. The bottom plot again shows that the latency in the system was also maintained below the latency limit for the majority of the time. The few cases where the latency limit was exceeded again correspond to rapid decreases in network bandwidth.

Using this control strategy the user would be able to run the system on any given TCP network with known, unknown or even variable network characteristics with good confidence that the system would be able to adhere to the real-time latency constraints set by the user while maximizing the data resolution for the wireless network being used.

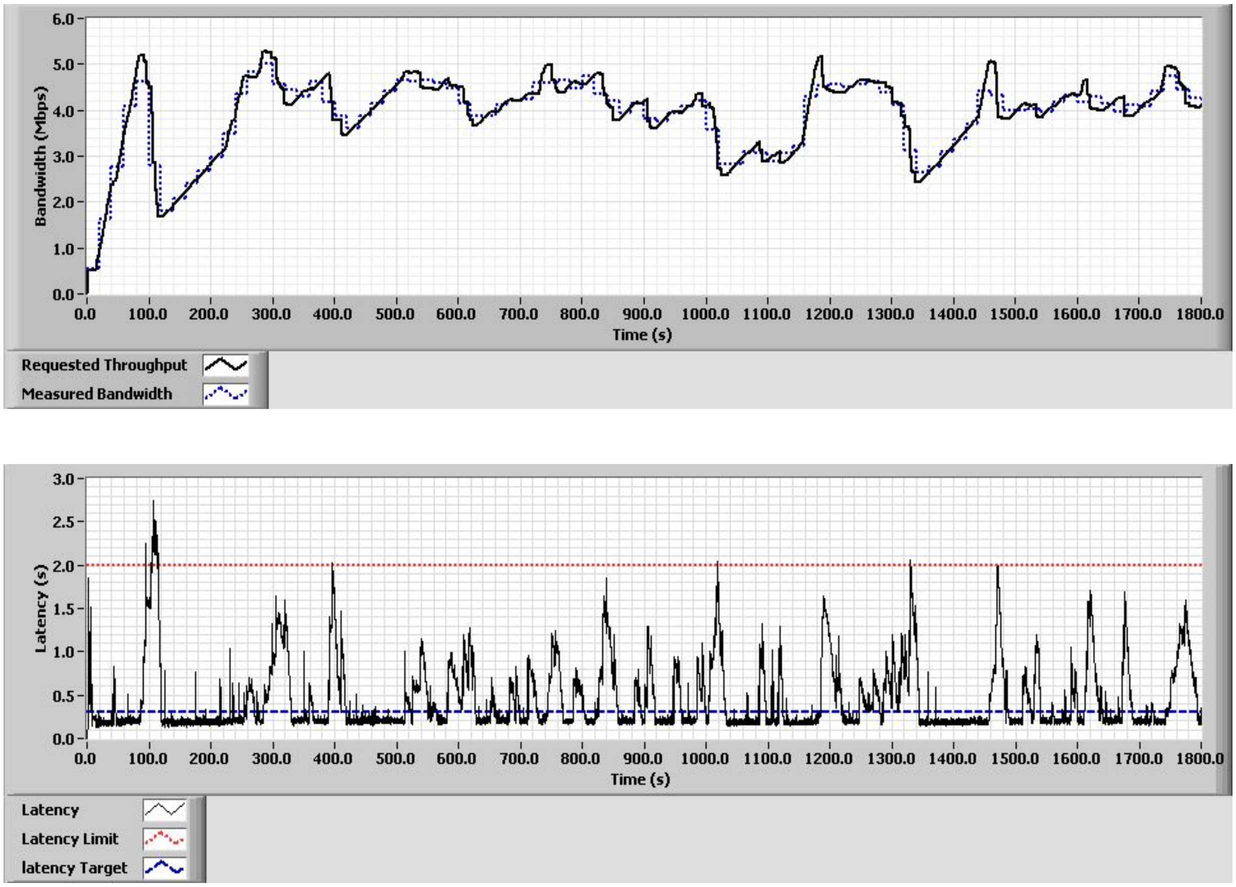


Figure 14. Uncontrolled wireless network test (Top plot: Requested System throughput and network bandwidth; Bottom plot: Data latency)

4. CONCLUSION REMARKS

The challenge in conducting real-time high bandwidth experimentation or control using wireless TCP/IP networks is that the latency requirements of these applications may not be reliably met due to various limitations and the unpredictable nature of wireless networks. The developed methods allow for dynamic adjustments to be made to the throughput of the system by enabling real-time adjustment of sensor data resolution (without drastic loss of sensor data quality). This allows the operator to closely match the throughput of the sensor data to the bandwidth available on the wireless TCP network thereby utilizing the network resources more efficiently while minimizing the latency in the system. The system was also demonstrated using a real life remote data acquisition application.

In addition to remote data acquisition the presented method has many potential applications in a variety of fields, such as education, business, medicine, and entertainment.

5. ACKNOWLEDGMENTS

This work was supported by ONR Award # N00014-05-1-0632: Navy Force Protection Technology Assessment Project. Authors acknowledge support and collaboration from the Maritime Security Laboratory at Stevens Institute of Technology.

6. REFERENCES

[1] L.G.Kim and P.T. Sakai, , **IEEE 802.11 Wireless Local Area Networks** Crow, B.P.; Widjaja, I.; -Communications Magazine, IEEE

[2] U. Varshney and R. Vetter, **Emerging Mobile and wireless Networks**, Communications of the ACM, 2000

[3] J. Ander, **Radio Resource Management in Future Wireless Networks: Requirements and Limitations**, Communications Magazine, IEEE, Vol.35, Iss.8, Aug 1997, Pages:30-36

[4]J. Li, C. Blake, D. Couto, S. J. Lee and H. I. Morris, **Capacity of Ad hoc Wireless Networks,** 7th Ann. Int. Conf. on Mobile Cmpting and Ntwking; Rome; Italy; pp. 61-69. 2001

[5] Gavin; Vaidya, Nitin, **Analysis of TCP Performance over Mobile Ad hoc Networks**, Holland, WIRELESS NETWORKS. Vol. 8, no. 2-3, pp. 275-288. 2002

[6] Y. Tian; K. Xu and N. Ansari, **TCP in Wireless Environments: Problems and Solutions**, Communications Magazine, IEEE, Vol.43, Iss.3, March 2005, pages: S27-S32

[7] Z. Fu, X. Meng; S. Lu, **How Bad TCP Can Perform in mobile Ad hoc Networks**, Computers and Communications, 2002, Proceedings, ISCC 2002, Pages: 298- 303

[8] J. G. Kim and M. M. Krunz, **Bandwidth Allocation in Wireless Networks with Guaranteed Packet-loss Performance**, Networking, IEEE/ACM Transactions on, Vol.8, Iss.3, Jun 2000, Pages:337-349

[9] National instruments, PCI-6123 DAQ data Sheet, http://www.ni.com/pdf/products/us/043918301101dlr.pdf

[10]Bandwidth Controller Standard Network Manager Software, http://bandwidthcontroller.com

[11] Jerri, A.J., **The Shannon sampling theorem—Its various extensions and applications: A tutorial review**, *Proceedings of the IEEE* , vol.65, no.11, pp. 1565-1596, Nov. 1977

[12] Tian He, Brian M. Blum, Qing Cao, John A., Stankovic, Sang H. Son, Tarek F. Abdelzaher, "**Robust and Timely Communications Over Highly Dynamic Sensor Network", *Springer Journal of Real-time Systems***, *Special Issue: Real-Time Wireless Sensor Networks*, v 37, n 3, December, 2007

[13] Sivaradje, G., and Dananjayan, P., **Efficient Resource Allocation Scheme for Real-time Mpeg Video Traffic Over ATM Networks,** *IEEE International Conference on Communications Systems* **(**IEEE ICCS 2002)