Human-Guided Autonomy for Acoustically Tethered Underwater Vehicles

Chris Murphy
Dept. of Applied Ocean Physics and Engineering
Woods Hole Oceanographic Institution
Woods Hole, MA 02543 USA
Email: cmurphy@whoi.edu

Hanumant Singh
Dept. of Applied Ocean Physics and Engineering
Woods Hole Oceanographic Institution
Woods Hole, MA 02543 USA
Email: hsingh@whoi.edu

Abstract— Recent under-ice Arctic AUV expeditions [1] have shown that polar operations require careful coordination between subsea vehicles and surface ships during vehicle recovery. In addition, the complexity of AUV launches and recoveries encourages telemetering AUV science data to allow scientists greater decision-making power while an AUV dive is underway. Reliable underwater communication, however, imposes strict bandwidth limitations. This paper presents a method for summarizing scalar vehicle science and state telemetry over low-bandwidth acoustic links, along with a topside display for presenting the data to surface observers.

I. Introduction

Myriad ROVs, ranging from shallow-water commercial models to the *JASON II* [2] ROV developed at the Woods Hole Oceanographic Institution, allow human operators to explore the depths by proxy; a joystick and computer monitors replacing normal sensory methods. Scientists and engineers can observe imagery, video, and science data in realtime as it is telemetered up through a tether to the surface ship. Viewing this data, they can begin to form hypotheses from new observations, develop plans for upcoming dives, and even alter the current dive to match developments. However, the very same tether puts a number of limitations on the capabilities of the ROV. Bound to the surface ship, an ROV cannot operate any distance further from the ship than the length of cable available. This means that the surface ship cannot leave the ROV for any reason. It also hampers or precludes operation in environments where ship motion is severely limited, such as under Arctic ice caps. Finally, ROVs typically possess minimal innate control logic; thus they must constantly have one or multiple human operators available to control the ROV's actions.

In contrast, AUVs require no physical surface tether. This allows AUVs to reach areas that are inaccessible to ROVs, or be left for hours to days before recovery. However, AUVs are often preprogrammed; missions are typically described in a high level language as a set of waypoints and leave few, if any, mission planning decisions to the vehicles control systems. As Christopher von Alt noted in a 2003 whitepaper,

> in general [AUVs do not] use sensor data obtained during a mission to make them more successful and/or reliable. Sensor information is recorded. It is

not processed and used to provide the vehicle with the ability to adapt, and change its current objective; it is simply recorded for future analysis. [3]

While progress continues to be made towards higher levels of autonomy in AUVs, training an AUV to alter missions in a way that actually achieves high-level mission objectives is a task-dependant and very complex problem. Ideally, AUVs could benefit from the domain and application-specific knowledge possessed by the scientists and engineers miles above on surface ships. If an AUV could communicate enough information to the surface-ship about sensor readings and vehicle state, human operators could adapt mission goals, much as with an ROV. Unfortunately, the ocean imposes severe limits on underwater communications, mandating careful use of each bit of telemetry. This paper presents a complete system for communicating scalar telemetry to surface observers using commercially available acoustic modems operating at ultra low bit rates of $O(10)$ to $O(100)$ bits per second.

II. Underwater Communications

While typical surface or air-based robots might communicate using high-frequency electromagnetic signalling, such as radio modems or WiFi, electromagnetic radiation is quickly dispersed by water. The new Hybrid ROV being developed at WHOI, Nereus [4], has the potential to deliver surface operators telemetry at very high rates by using a single strand of fiber-optic cable instead of acoustic communication. While this is an exciting development in the field of underwater robotics, acoustic communication remains the only long-distance and wireless underwater solution.

The ocean environment presents numerous challenges for acoustic communication, including low available bandwidth and large propagation delays [5]. These challenges are made worse by operating over long distances [6] and by environmental conditions such as seafloor makeup and water depth. AUV and surface ship noise transmit directly into the channel, further exacerbating the problem. As a result, use of long-range underwater communication is characterized by extremely low effective bandwidth, high latency, and frequent packet loss.

To accomodate the peculiarities of the medium, channel coding methods with high rates of error-correction are typ-

ically employed. While underwater acoustic communications has achieved rates up to hundreds of kilobits per second [7], reliable acoustic communications over long distances currently requires the use of low-rate communications with high error tolerance, such as frequency-hopping frequency shift keying (FH-FSK) or highly error-corrected phase shift keying (PSK). In addition, AUVs may rely on acoustic navigation schemes such as LBL [8] or USBL. Since the ocean is a shared broadcast medium, time-multiplexing of the channel for navigation or communication with other vehicles may be required, which lowers effective bit-rates further.

A. Hardware

AUVs are often severely limited by power constraints, as all power must be brought down with the AUV or generated underwater. Most AUV therefore obtain power from large on-board battery packs. Acoustic modems designed for AUVs should therefore ideally use minimal power. Seabed-class AUVs [9] use the WHOI Micro-Modem [10] for communication and navigation [11]. To minimize power usage, the Micro-Modem has fixed firmware and functionality; this allows it to use only 10 Watts while transmitting and only 0.08 Watts while receiving [12]. While at sea, it is crucial that telemetry can be adapted to meet potentially changing needs of specific missions. Thus, software-implemented encoding solutions are often pursued instead of modifying lower-level processing within the Micro-Modem.

The WHOI Micro-Modem provides Media Access Control (MAC), and uses low frequency bandwidth to allow for multiple senders and receivers. It is capable of sending one 256 bit FH-FSK packet in slightly over 3 seconds, or one 1536 bit error-tolerant PSK packet in slightly over 6 seconds, delivering an effective bit-rate between 80 and 256 bits per second. Summarizing data for transmission at such low bit-rates presents a significant hurdle. As a result, current telemetry is often quite limited. Seabed-class and REMUS AUVs currently make use of the Compact Control Language (CCL) [13], which was developed at WHOI to meet the needs of an AUV.

B. Compact Control Language (CCL)

CCL provides a data-formatting standard for AUV to AUV, and AUV to surface-ship, communications. The standard describes a number of 256 bit packets which can be used for file transfer or for transmitting vehicle state, salinity data, bathymetry estimates, and other oceanographic data [14]. Each of these packets is designed to be self-contained; for example, the MDAT_BATHY (bathymetry) packet contains measurements of depth, altitude, latitude and longitude from three distinct locations. No other packets are required to make sense of the data contained within the packet. To provide rudimentary compression, data are typically requantized to varying degrees to fit into an integer number of bytes. Seabed-class AUVs use a simple CCL packet containing the vehicle's location, current mission goal, and health, as shown in Table I. The AUV is typically configured with a simple TDMA cycle

to telemeter packets to the surface ship at regular intervals. Each field of the CCL packet contains the most recent sample or estimate for that source.

TABLE I

CONTENTS OF STANDARD SEABED CCL PACKET. RECENTLY, UNUSED BITS HAVE BEEN ADOPTED FOR A VARIETY OF MISSION-SPECIFIC DATA.

Data	Bits	Precision
Packet type code	8	N/A
Position (X,Y)	48	1 m
Depth	16	0.1 m – 0.5 m
Altitude	16	0.01 m
Heading	8	1.4°
Goal Position (X,Y)	48	1 m
Goal Depth	16	0.1 m – 0.5 m
Goal ID or Error Code	16	N/A
Unused	80	N/A

While this strategy is sufficient to provide a surface ship an indication of the AUV's location and goal, the data telemetered may be strongly aliased due to the low sample rate. Aliasing of vehicle heading is particularly likely, as an AUV without the ability to control heading (such as caused by an inoperational thruster) may spin at a much higher rate than the Nyquist frequency of the telemetry allows it to represent. An example of this is shown in Figure 1, using heading data from the JAGUAR AUV [15] acquired during the Arctic Gakkel Vents Expedition (AGAVE) [16] as the AUV spiraled to the seafloor.
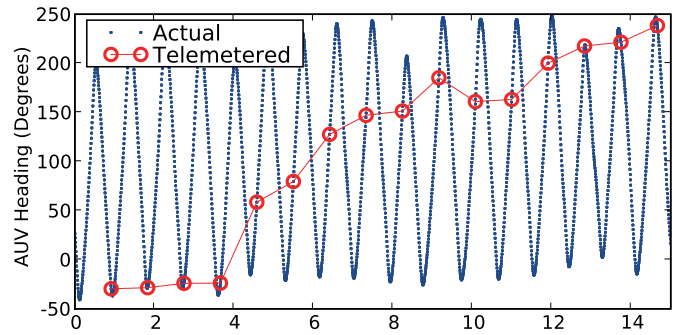

Fig. 1. Heading sensor aliasing during the Arctic Gakkel Vents Expedition. Red circles show telemetry to be sent to a surface ship with a 55 second long TDMA cycle. X Axis is in minutes.

Furthermore, the low sampling resolution of the telemetry in *time* may translate directly to an unacceptably low sampling resolution in *space*. An AUV with a horizontal speed of $0.5 \frac{m}{s}$, telemetering data every 60 seconds, will only provide one sample for every 30m of forward travel. To alleviate both resolution and aliasing concerns, data can be decimated appropriately with an anti-aliasing filter. Rather than use the natural sampling frequency of an instrument, it may make more sense to subsample an instrument's data to meet some other desired criterion. For missions with a desired horizontal spatial sampling resolution, a sampling frequency could trivially be calculated from a nominal or maximum vehicle speed.

Alternatively, summary statistics (such as a running mean or variance) can be calculated and telemetered.

This strategy, sending summary statistics, was used during a 2008 research cruise to the Southern Mid-Atlantic Ridge (SMAR) to telemeter some scientific data to the surface while the AUV continued its tracklines. The "Unused" bits in the MDAT_STATEXY CCL packet (shown in Table I) were allocated to transmitting summarized Optical Backscatter (OBS) and Reduction Potential (Eh) science data. OBS data was median filtered to remove noise, and anti-aliased before being telemetered. Eh data acquired since the last telemetry packet was low-pass filtered with a Butterworth filter, differentiated, and the variance calculated. While these areas of high variance were found to correlate well with areas of interest in the raw data, the telemetered value is difficult to interpret in familiar units and terms. To truly allow scientists a clear understanding of what the AUV is sensing, time-series plots and geo-referenced plots at high sampling rates are necessary. Ideally, scientists should be presented with data at high enough resolution that the full-resolution data can be used for brief confirmation if the AUV is recovered, but is not necessary for decision-making.

III. TELEMETRY

Many standard oceanographic sensors, such as an Eh or CTD sensor, emit a time series of one or multiple scalar floating-point values as their data product. A simple temperature sensor operating at 10Hz with only a single scalar measurement value will produce 19.2 kilobits of 32-bit floating-point data per minute. AUV position information may also be required for telemetry to be useful; if the surface ship has no external method of obtaining the vehicle position, such as a USBL system, the AUV location, depth, and altitude may also need to be transmitted to the surface.

Providing all of this data at full resolution would require orders of magnitude more bandwidth than is currently avaiable via robust acoustic transmission. Instead, most AUVs currently provide very limited telemetry to surface observers, and store science data until the vehicle is recovered. Providing this data to surface-based observers would give earlier insights into the AUV's discoveries, but clearly requires more advanced compression.

A. Lossless Compression

There exist an entire alphabet of general-purpose lossless compression algorithms, such as LZ77 [17], Deflate [18], LZW [19], and the BWT [20]. These algorithms form the basis for a number of generic compression utilities; *gzip*, *winzip*, and *pkzip* utilities make use of the Deflate algorithm, while *bzip2* uses the BWT algorithm. As early as 1996, Eastwood et al. looked at the efficacy of these techniques, and proposed a couple of heuristics for use when transmitting data over an acoustic modem. [21] While these algorithms perform well with plain text and other widely used document formats, they do not perform particularly well on floating-point

scientific data. Welch addresses this directly in his notes on the performance of LZW [19], saying:

> Arrays of floating point numbers look pretty much like white noise and so they compress rather poorly. The fractional part is a nearly random bit pattern. ... Some floating point arrays expand by 10 percent going through the compression algorithm, when the exponents vary widely.

As a result, special-purpose approaches designed especially for compressing floating-point scientific data have been developed [22]–[24]. However, even the most recent lossless algorithms yield on the order of 2:1 compression for high entropy time-series science data [22], suggesting that lossy compression is likely to be required for the high demands of acoustic telemetry. This is not necessarily detrimental, as the goal of low-bandwidth AUV telemetry should be to supply operators with a rough sketch of the AUV's state and environment in as many modes as possible, not necessarily to provide observers with full-resolution results.

B. Lossy Compression

Most lossy data compression schemes use a similar pattern to obtain reasonable compression levels, as shown in Figure 2. First, data is encoded into a new domain where it is believed to have a more easily compressed representation, using methods such as the Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT). When encoding a sinusoid, for example, the results of compressing a single magnitude and phase in DCT space are likely to be much better than the results of encoding an entire series of discrete-time samples. Next, the resultant data is either scalar or vector quantized before being re-encoded with some form of entropy encoding [25].
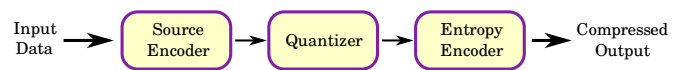

Fig. 2. Standard lossy signal encoder. Data is first compressed using a source encoder then quantized, then entropy coded. Graphic modified from [25].

IV. WAVELET COMPRESSION OF OCEANOGRAPHIC DATA

To compress time-series data to a manageable size, we make use of the DWT to do a simple form of wavelet compresssion. The method employed is described by Donoho et al. as being especially appropriate for functions that are "piecewise-smooth away from discontinuities" [26]. This is an apt description for the sort of a signal that would be emitted by an oceanographic sensor, such as an AUV's Eh or OBS sensor, as it comes upon a feature of interest. Vetterli provides an excellent introduction to wavelet compression, as well as a comparison to Fourier based compression [27]. Since wavelets are localized in time, wavelet coefficients will be larger near discontinuities and abrupt changes.

A. Source Encoder

The Discrete Wavelet Transform generates two sets of coefficients, approximation and detail coefficients, each time it is applied to a dataset. The transform can be iteratively applied to the approximation coefficients to generate a smaller number of approximation coefficients and a larger number of detail coefficients. The transform itself is lossless and can easily be reversed via the inverse DWT, much like the FFT in Fourier space. Approximation coefficients can be thought of as a low-fidelity representation of the signal, with the detail coefficients filling in the details. Each detail coefficient is localized in time, as well as having a scale associated with it. For a well-written and more formal introduction to wavelets, DeVore and Lucier provide an excellent reference [28].

Actual source encoding of the time-series data is quite straightforward. First, the DWT is iteratively applied to the time-series data, resulting in a multilevel wavelet decomposition. The result of this decomposition is a small number of approximation coefficients, and a large number of detail coefficients. For smooth time-series data with few discontinuities, the detail coefficients will be low in magnitude, with interspersed large magnitude coefficients near discontinuities. This sparsity turns out to be of great benefit during quantization and entropy coding.

B. Quantization

For simplicity, a final fixed data size based upon the requirements of a pre-set number of packets and TDMA schedule can be set for the compressed data. A simple schedule, such as the one in Table II, can provide generous cycle lengths, an LBL ping each minute, and two standard 'State' packets for fallback while still supporting a surprising number of sensors.

The fixed data size governs the level of quantization, and determines the quality of the compression. While care should be taken when quantizing the approximation coefficients, which represent a low-level approximation to the signal, detail coefficients represent time-localized changes to the signal which can be quantized.

TABLE II

SAMPLE TDMA SCHEDULE FOR DEEP CHEMICAL SENSING

#	Cycle Name	Cycle Length	#	Cycle Name	Cycle Length
1	X Position	14 sec.	10	Potential Temp.	14 sec.
2	X Position	14 sec.	11	Potential Temp.	14 sec.
3	Y Position	14 sec.	12	Redox Potential	14 sec.
4	LBL Nav.	11 sec.	13	LBL Nav.	11 sec.
5	Y Position	14 sec.	14	Redox Potential	14 sec.
6	Depth	14 sec.	15	Opt. Backscatter	14 sec.
7	Depth	14 sec.	16	Opt. Backscatter	14 sec.
8	LBL Nav.	11 sec.	17	LBL Nav.	11 sec.
9	CCL State	14 sec.	18	CCL State	14 sec.
Total schedule length:			240 seconds = 4 minutes		
Rate for each timeseries:			$\frac{64 \text{ bytes}}{4 \text{ min.}} = 2.13 \frac{\text{bits}}{\text{sec.}}$		

Approximation coefficients are transmitted as simple 32-bit floating-point values, though the number of coefficients can be minimized by selecting an appropriate wavelet basis function and by iteratively applying the wavelet decomposition the maximum number of times. Detail coefficients are quantized as either half-precision (16-bit) floating point numbers or 16-bit fixed point numbers depending on the application.

Each 16-bit quantized coefficient also requires several bits of storage for position information as described in the next section. As a fixed amount of space is available for the compressed data, the number of detail coefficients which can be stored, N, is easily calculable. The top N detail coefficients are then entropy coded, and every other detail coefficient is zeroed. This results in a very sparse (and therefore easily entropy-codable) representation. The signal will be de-noised as a side benefit; discarding low-magnitude wavelet coefficients is an effective form of noise reduction [27].

C. Entropy Coding of Very Sparse Data

Entropy encoders are simple forms of lossless compression, designed to encode low-entropy data in less space than would otherwise be required. Two common techniques include Huffman Coding [29] and Run-Length Encoding (RLE). Huffman coding requires that the encoder and decoder share a decision tree showing the mapping of bits to symbols. This can either be shared ahead of time, such as a mapping based on the frequency of letters in general english text, or calculated optimally for a given dataset and transmitted along with the data. Given that the data statistics will likely be unknown beforehand and the transmitted data for AUV telemetry will be quite short, the overhead of sending a Huffman tree may be as large as (or larger than) the data itself.

RLE takes advantage of repeated sequences of symbols within data to compress the data. This is often done by providing some method of referring to previous data within the stream. As the encoded data in this case will be incredibly sparse, a modified RLE-like strategy is used.

First, the DWT approximation coefficients are transmitted as 32-bit floating-point values. The encoding wavelet is selected to minimize the number of approximation coefficients, as each must be encoded with relatively high fidelity to avoid quantization errors. The total number of coefficients is transmitted, and used to calculate the number of bits, M, needed to represent the location of any coefficient in the list. The rest of the encoded stream consists of an unsigned integer offset using M bits, followed by a quantized detail coefficient. Finally, a 24 bit time code is transmitted along with the packet in our current implementation. The time code contains the time of day of the final sample in the time series, measured in 20ths of a second, along with a quantized representation of the data sampling rate. These two numbers allow easy decoding of the time associated with the data. The time of day could be quantized much further, provided there were guarantees on the age of transmitted data (such as that it was no older than the time between TDMA cycles). The sampling rate could be omitted altogether if sensor sampling rates are known beforehand.

Including it, however, offers the option for the AUV to encode a decimated version of a longer time-series in environments with heavy packet loss, providing redundancy.

V. Geo-Referenced Display

AUV telemetry is most useful when geo-referenced to the other science data which has been acquired in the area. GeoZui3D, developed at the University of New Hampshire's Center for Coastal and Ocean Mapping, provides a powerful 3D interface for reviewing captured data [30]. Kinsey and Whitcomb at Johns Hopkins University have developed DVLNAV for monitoring Remotely Operated and Human-Occupied Vehicles [31]. In 2005, a system for real-time 3D monitoring of ROV's and HOV's was presented, developed as a merger of GeoZui3D and DVLNAV's capabilities [32]. To provide scientists with a simple to use display for exploring the AUV's science data, we have developed a Python-based dataserver which decodes our wavelet-encoded telemetry into a more workable format. Position telemetry is interpolated to the time-axis of each other dataseries that has been telemetered up. The server then stores the geo-referenced sensor data in a hierarchical database file, and provides it via a web server in KML [33] format to multiple clients simultaneously. Our system's primary goal is to provide a simple interface to the real-time data, understandable by scientists without a computer science background, capable of running on a variety of operating systems. To achieve this, the system builds on pre-existing geodesy tools such as *Google Earth*.

KML clients that support "Network Links", such as *Google Earth* or *ArcGIS Explorer*, will continue to receive updated data even as users explore the telemetry. Telemetry is presented as an easy-to-understand color coded scatter plot on the map, and each point also has a timestamp to allow easy navigation or playback. Using KML for the telemetry format also allows the tools being used for data analysis on the ship to be reused to provide mission data to the general public afterwards, as part of outreach efforts.

VI. Results

To evaluate the performance of the system in extreme circumstances, a particularly complex section of Eh data from a dive on the Southern Mid-Atlantic Ridge was wavelet encoded, along with vehicle state information. The raw sensor values and compressed results are shown in Figure 3. During this dive, the AUV had lost control due to a broken propellor. This resulted in the AUV spinning and randomly walking across the seafloor for a prolonged period of time. The time-series data was divided into sections of time-series telemetry 4 minutes long, each of which was wavelet compressed and uncompressed in turn. The simulated transmission followed the TDMA schedule shown in Table II.

Figure 4 shows in greater detail the results of compressing two hours of Reduction Potential (Eh) data acquired during a cruise to the Southern Mid-Atlantic Ridge to varying degrees. The plot shows the results of sub-sampling the Eh data to yield the same number of bytes for comparison, along with a
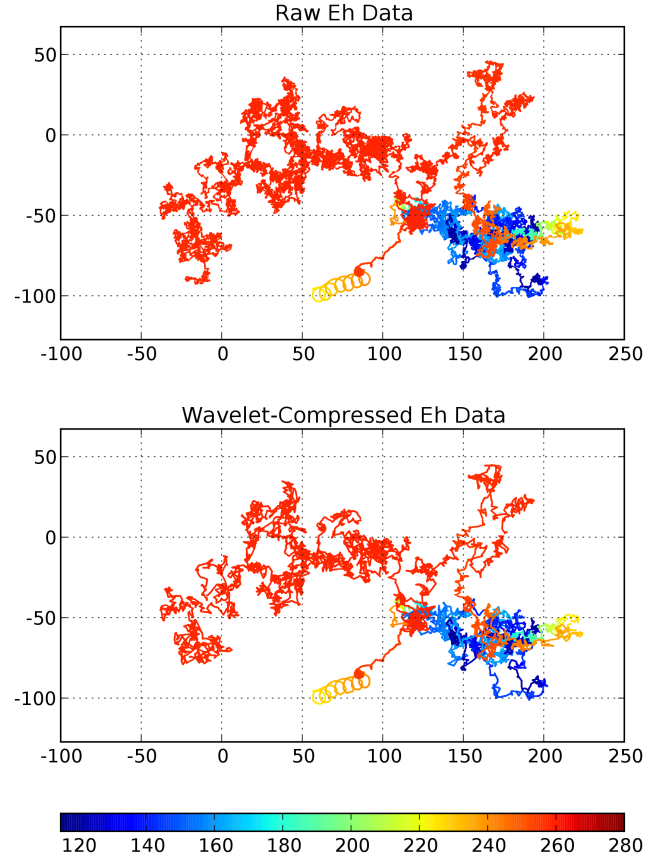

Fig. 3. Raw sensor values on top, versus wavelet-compressed Eh data. The X and Y positions were telemetered up independently, also using the same wavelet compression techniques. Both axes are in meters.

b-spline interpolation of the sub-sampled points. Using only 60 bytes, the wavelet compressed version shows two distinct dips in the value of the Eh sensor; in 120 bytes, the magnitudes of the drops are approximately correct. With 240 bytes, a low-resolution version of the signal is available, and by 360 bytes almost all the visible detail has been captured. Even operating in low-bandwidth FH-FSK mode, the WHOI Micro-Modem could transfer this two-hour section of telemetry in less than a minute. In contrast, at 60 bytes the representation yielded by sub-sampling is quite poor, and by 360 bytes the sub-sampled data still does not represent the second 'dip' in the data as two distinct troughs.

Once telemetry is received on the surface and decoded, the telemetry is forwarded to the data server for consumption by any KML clients. An example client interface browsing the provided data stream is shown in Figure 5. The interface shown is Google Earth; users can scan backwards and forwards in time, as well as switch between sensors. Additional geo-referenced information can be added by the user, or the data server, to be displayed along with the telemetry. This may be a single marked point, or more complicated data such as a multibeam bathymetric map or mission tracks.
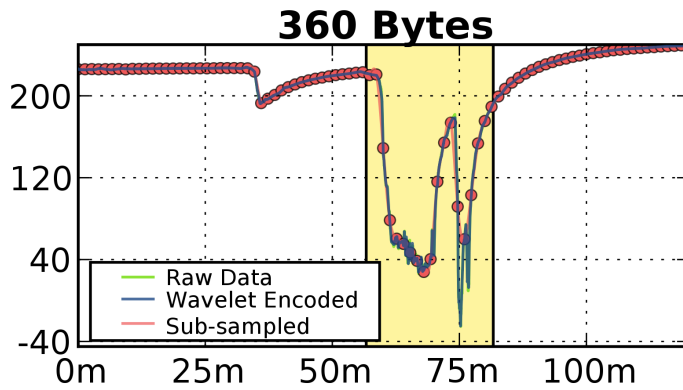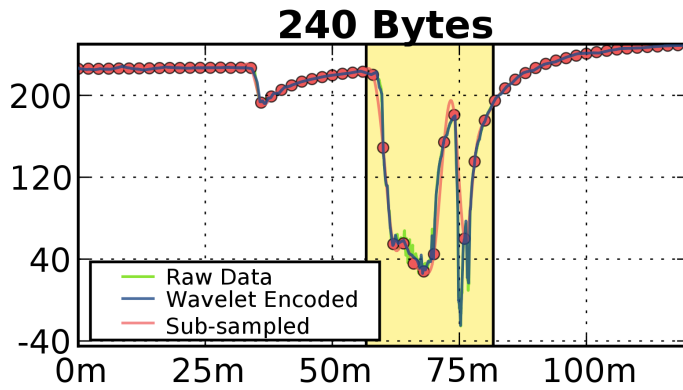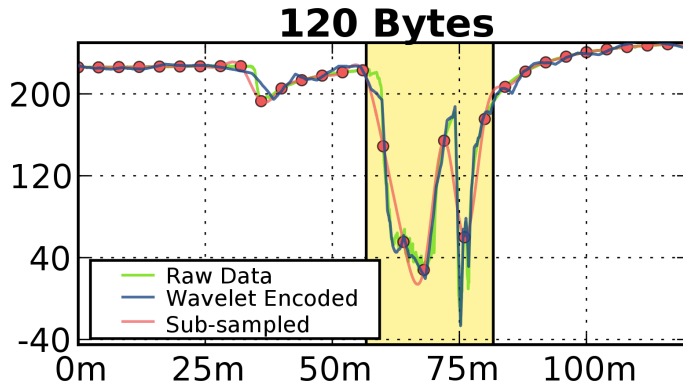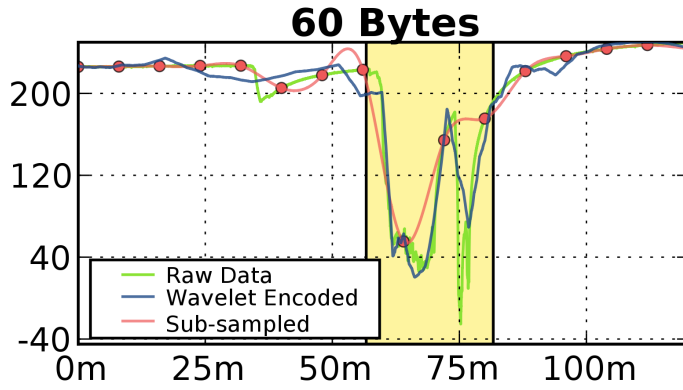
(a) Full time-series Eh data
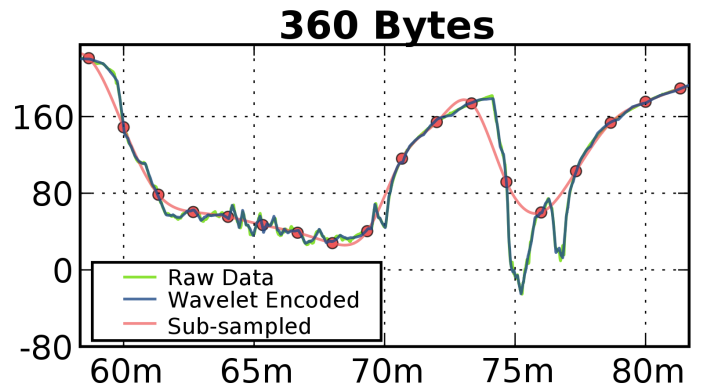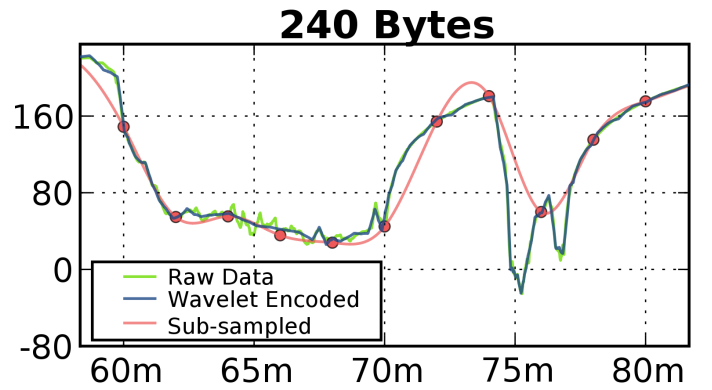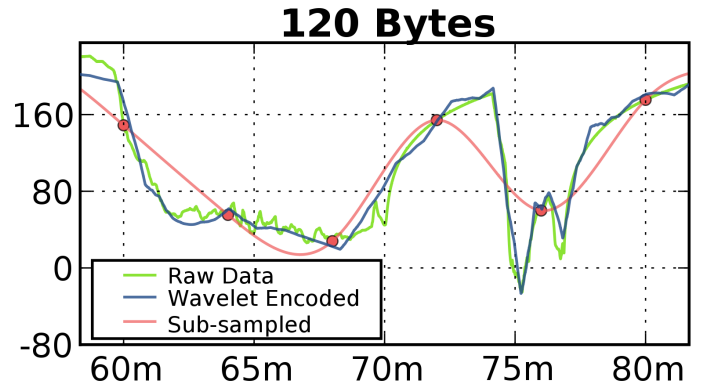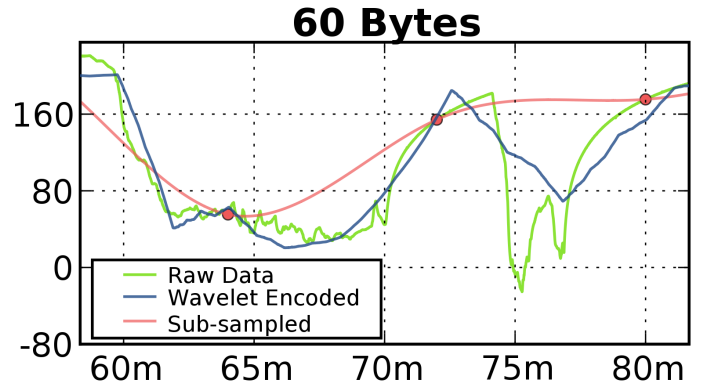
(b) Detail view of time-series Eh data

Fig. 4. Results of wavelet compressing Eh data compared to the raw data. Also shown is a sub-sampled version of the original data, sampled at a rate which would yield the same amount of data as the wavelet compressed version. The sub-sampled version has also been interpolated with a b-spline. The Daubechies 3 wavelet was used for performing the Discrete Wavelet Transform.
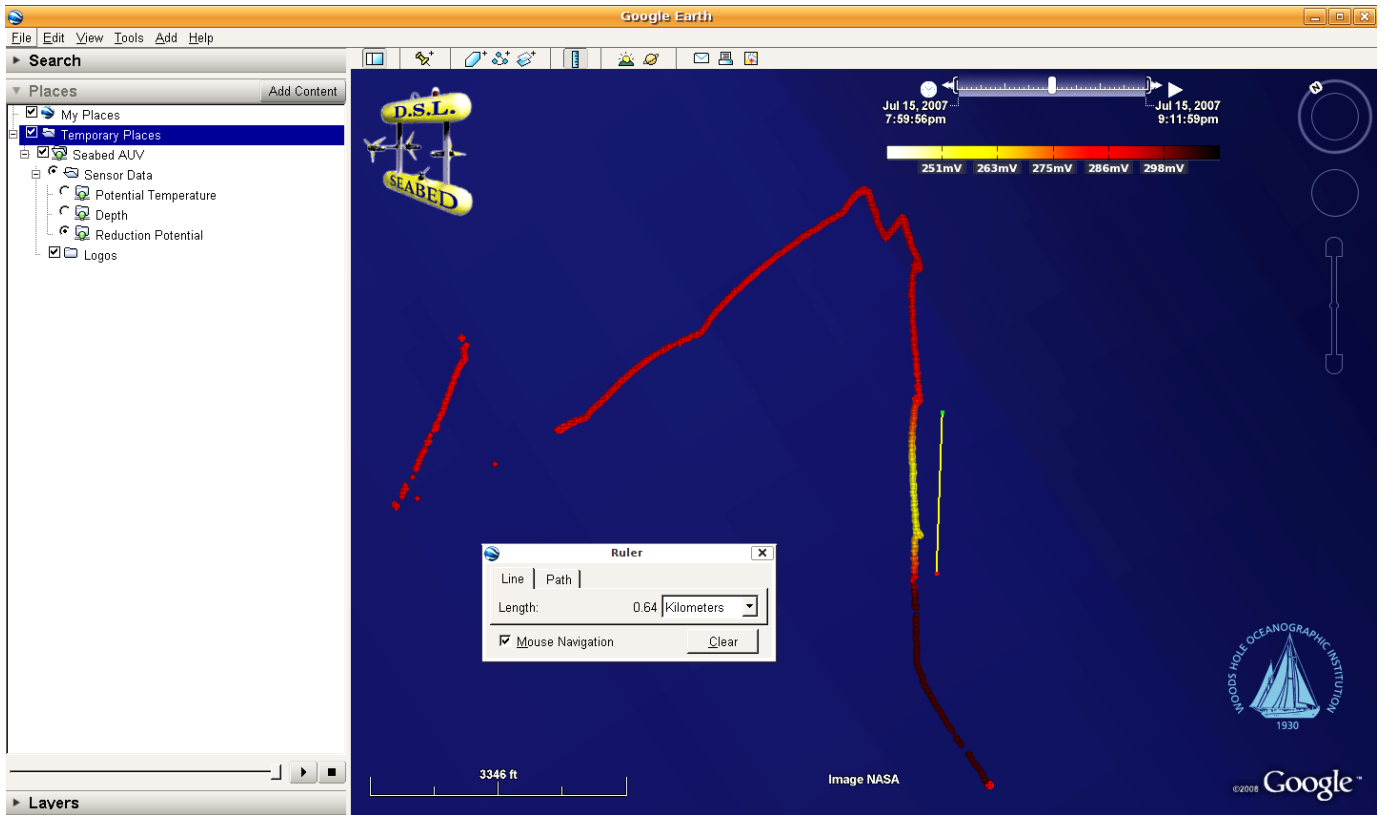
Fig. 5. Streaming Eh telemetry from a volcanically active portion of the Arctic Ocean's Gakkel Ridge [16] being reviewed in Google Earth. Users can scan through time, view data from different sensors, or annotate it. Shown is the use of a ruler to measure the length of an interesting section of Eh data.

VII. CONCLUSION

In this paper we present a complete system for compressing and reviewing time-series oceanographic data. This system has many opportunities for expansion, and is still under development. Recent developments include the inclusion of state telemetry from multiple AUVs, along with position information for any surface vehicles. Each of these data sources can be presented simultaneously, as shown in Figure 6. We continue to expand the capabilities of the topside monitoring software, which is now capable of tracking multiple AUVs and surface vehicles simultaneously.

In addition, there is extensive work to be done to further investigate the lossy compression of oceanographic data. There are many families of wavelets, each of which can be used to perform the DWT. In this paper, the Daubechies 3 wavelet was used exclusively. Calculating the optimal wavelet for compression, based upon previously captured data, could improve compression results even further.
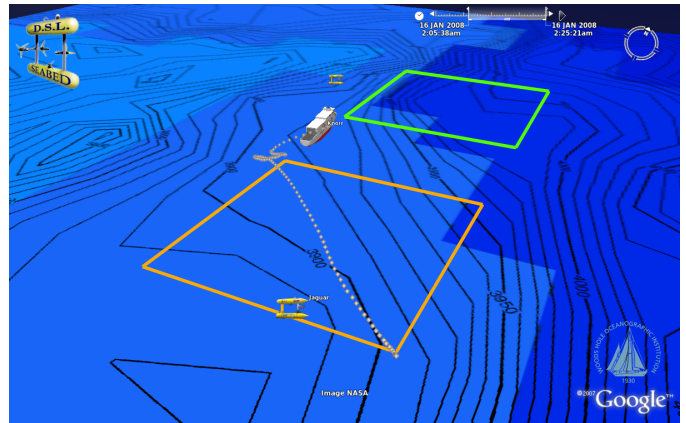
Fig. 6. Oblique view of live telemetry from the ship and two AUVs during mission on the Southern Mid-Atlantic Ridge. Also shown is Bathymetry acquired by the ship's multibeam system.

REFERENCES

[1] C. Kunz, C. Murphy, R. Camilli, H. Singh, J. Bailey, R. Eustice, C. Roman, M. Jakuba, C. Willis, T. Sato, K. ichi Nakamura, and R. A. Sohn, "Deep sea underwater robotic exploration in the ice-covered Arctic ocean with AUVs," in *Proc. IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, Nice, France, Sept. 2008.

[2] D. R. Yoerger and J. B. Newman, "Jason: An integrated approach to rov and control system design," in *Proceedings ROV 1986, MTS*, Washington, DC, 1986.

[3] C. von Alt, "Autonomous Underwater Vehicles," in *Autonomous and Lagrangian Platforms and Sensors*, 2003. [Online]. Available: http://www.geo-prose.com/ALPS/white_papers/alt.pdf

[4] A. Bowen, D. Yoerger, and L. Whitcomb, "Hybrid ROV for 11,000 Meter Operations," in *Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies*, Apr. 2007.

[5] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater Acoustic Sensor Networks: Research Challenges," in *Ad Hoc Networks*. Elsevier, Mar. 2005, vol. 3, no. 3, pp. 257 – 279.

[6] M. Stojanović, "On the relationship between capacity and distance in an underwater acoustic communication channel," in *SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, Oct. 2007, pp. 34 – 43.

[7] ——, "Recent advances in highspeed underwater acoustic communications," *IEEE Journal of Oceanic Engineering*, vol. 21, no. 2, pp. 125–136, Apr. 1996.

[8] P. H. Milne, *Underwater Acoustic Positioning Systems*. Houston, TX: Gulf Publishing Co., 1983.

[9] H. Singh, A. Can, R. Eustice, S. Lerner, N. McPhee, O. Pizarro, and C. Roman, "Seabed AUV offers new platform for high-resolution imaging," *EOS, Transactions of the AGU*, vol. 85, no. 31, pp. 289,294–295, Aug. 2004.

[10] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball, "The WHOI Micro-Modem: An Acoustic Communcations and Navigation System for Multiple Platforms," in *Oceans 2005, Proceedings of the MTS/IEEE*, vol. 2, 2005, pp. 1086 – 1092. [Online]. Available: http://acomms.whoi.edu/60x%20Reports%20and%20Papers/MicromodemOceans2005.pdf

[11] S. Singh, M. Grund, B. Bingham, R. Eustice, H. Singh, and L. Freitag, "Underwater Acoustic Navigation with the WHOI Micro Modem," in *Proc. IEEE/MTS Oceans Conference and Exhibition*, Boston, USA, 2006.

[12] W. A. C. Group, "WHOI Acoustic Communications: Micro-Modem," July 2008, http://acomms.whoi.edu/umodem/.

[13] R. Stokey, L. Freitag, and M. Grund, "A Compact Control Language for auv acoustic communication," *Oceans 2005 - Europe*, vol. 2, pp. 1133–1137, June 2005.

[14] R. P. Stokey, "A compact control language for autonomous underwater vehicles," Woods Hole Oceanographic Institution," Specification, Apr. 2005, http://acomms.whoi.edu/40x%20Specifications/401100%20Compact%20Control%20Language/CCL%20April%202005%20Public%20Release%201.0.pdf.

[15] C. Kunz, C. Murphy, H. Singh, C. Willis, R. A. S. nd Sandipa Singh, T. Sato, K. Nakamura, M. Jakuba, R. Eustice, R. Camilli, and J. Bailey, "Toward extraplanetary under-ice exploration: Robotic steps in the arctic," *Journal of Field Robotics Special Issue on Space Robotics*, submitted.

[16] R. Sohn, C. Willis, S. Humphris, T. Shank, H. Singh, H. Edmonds, C. Kunz, U. Hedman, E. Helmke, M. Jakuba, B. Liljebladh, J. Linder, C. Murphy, K. Nakamura, T. Sato, V. Schlindwein, C. Stranne, M. Tausenfreund, L. Upchurch, P. Winsor, M. Jakosson, and A. Soule, "Explosive volcanism on the ultraslow-spreading Gakkel ridge, Arctic Ocean," *Nature*, vol. 453, no. 7199, pp. 1236–1238, June 2008.

[17] J. Ziv and A. Lempel, "A universal algorithm for data compression," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 337–343, May 1977.

[18] L. P. Deutsch, "RFC 1951: DEFLATE compressed data format specification version 1.3," May 1996, status: INFORMATIONAL. [Online]. Available: ftp://ftp.internic.net/rfc/rfc1951.txt

[19] T. A. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, no. 6, pp. 8–19, June 1984.

[20] M. Burrows and D. J. Wheeler, "A block-sorting lossless data compression algorithm," Digital Equipment Corporation Systems Research Center, Research Report 124, May 1994.

[21] R. L. Eastwood, L. E. Freitag, and J. A. Catipovic, "Compression techniques for improving underwater acoustic transmission of images and data," in *Proceedings of MTS/IEEE OCEANS 1996: Prospects for the 21st Century*, Fort Lauderdale, FL, Sept. 1996.

[22] M. Burtscher and P. Ratanaworabhan, "High throughput compression of double-precision floating-point data," in *2007 Data Compression Conference (DCC'07)*, 2007.

[23] P. Lindstrom and M. Isenburg, "Fast and efficient compression of floating-point data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, Sept. 2006.

[24] P. Ratanaworabhan, J. Ke, and M. Burtscher, "Fast lossless compression of scientific floating-point data," in *2006 Data Compression Conference (DCC'06)*, 2006.

[25] S. Saha, "Image Compression - from DCT to Wavelets : A Review," *ACM Crossroads: Data Compression*, Spring 2000. [Online]. Available: http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html

[26] D. L. Donoho, M. Vetterli, R. DeVore, and I. Daubechies, "Data Compression and Harmonic Analysis," *IEEE Transactions on Information Theory*, vol. 44, no. 6, Oct. 1998.

[27] M. Vetterli, "Wavelets, approximation, and compression," *IEEE Signal Processing Magazine*, pp. 59 – 73, Sept. 2001.

[28] R. A. DeVore and B. J. Lucier, "Wavelets," in *Acta Numerica 1*. Cambridge University Press, 1992, pp. 1–56.

[29] D. A. Huffman, "A method for the construction of minimum-redundancy codes," in *Proceedings of the Institute of Radio Engineers*, vol. 40, no. 9, Sept. 1952, pp. 1098–1101.

[30] C. Ware, M. Plumlee, R. Arsenault, L. Mayer, and S. Smith, "Geozui3d: Data fusion for interpreting oceanographic data," in *MTS/IEEE Oceans 2001 Conference and Exhibition*, vol. 3, Honolulu, HI, Nov. 2001, pp. 1960–1964.

[31] J. C. Kinsey and L. L. Whitcomb, "Preliminary field experience with the DVLNAV integrated navigation system for oceanographic submersibles," *Control Engineering Practice*, vol. 12, no. 12, pp. 1541–1548, December 2004, invited Paper.

[32] S. C. Martin, L. L. Whitcomb, R. Arsenault, M. Plumlee, and C. Ware, "A system for real-time spatio-temporal 3-d data visualization in underwater robotic exploration," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, Apr. 2005.

[33] Open Geospatial Consortium, "KML Standard," http://www.opengeospatial.org/standards/kml/.