A State Observation Technique for Highly Compressed Source Coding of Autonomous Underwater Vehicle Position

Toby Schneider, *Member, IEEE,* Henrik Schmidt
Center for Ocean Engineering
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139
tes@mit.edu and henrik@mit.edu

***Abstract*—A novel technique is presented for using state observers in conjunction with an entropy source encoder to enable highly compressed telemetry of autonomous underwater vehicle (AUV) position vectors. In this work, both the sending vehicle and receiving vehicle or human operator are equipped with a shared real-time simulation of the sender's state based on the prior transmitted positions. Thus, only the innovation between the sender's actual state and the shared state need be sent over the link, such as a very low throughput acoustic modem. The distribution of this innovation can be modeled *a priori* or assembled adaptively. This distribution is then paired with an arithmetic entropy encoder, producing a very low cost representation of the vehicle's position vector.**

This system was analyzed on experimental data from the GLINT10 and AGAVE07 expeditions involving two different classes of AUVs performing a diverse number of maneuvers, and implemented on a fielded vehicle in the MBAT12 experiment. Using an adaptive probability distribution in combination with either of two state observer models, greater than 90% compression, relative to a 32-bit integer baseline, was achieved.

***Index Terms*—autonomous underwater vehicles, acoustic communications, robotic networks**

I. Introduction

A. Motivation

Users of mobile marine platforms such as autonomous underwater vehicles (AUVs) and unmanned surface vehicles (USVs) are one of the major beneficiaries of improved acoustic communication capabilities, since the need to move often precludes the use of fiber optic communication tethers. These vehicles are also becoming increasingly "intelligent"; they are outfitted with substantial computational ability and are capable of fulfilling complex mission components or entire missions autonomously; for examples, see [1]–[3].

For many types of AUV missions it is required or desirable for the vehicle (here, the *sender*) to transmit accurate and frequent vehicle position measurements to collaborating vehicles or a human operator (the *receiver*). For example, oceanographic missions require the position where sensor samples were taken, and collaborative target detection tasks require a history of positions to avoid unnecessary redundant coverage, or facilitate coordinated control maneuvers such as formation flying. Furthermore, as vehicle navigation decisions become increasingly automated, human operators desire increased assurance that their highly expensive vehicles are operating correctly and away from hazards.

This need for vehicle position knowledge can often consume much or all of the available acoustic link's throughput in fielded vehicles. In this paper, a system is devised that uses a matched state observer on the *sender* and *receiver* to reduce the position vector to a vector of differences from the modeled state. The probability distribution of these differences is modeled *a priori* or adaptively built from prior statistics. The resulting distribution is coupled with an arithmetic entropy encoder to provide highly compressed position vectors.

This work was funded by the Office of Naval Research (ONR) under projects N00014-08-1-0011 and N00014-11-1-0097.

B. Related Work

Much work has been done on understanding the physical channel for acoustic telemetry; see [4] for a review of the last decade. While the focus has been on error-free channel coding and transmission of datagrams, little has been published in the marine domain on source coding of underwater measurements as evidenced by [5] and a dearth of coverage of source coding in the major underwater networking review papers [6], [7].

One exception is Murphy's work [8], [9], in which he uses transform compression (e.g. the discrete wavelet transform) to source encode imagery and historical time series of scalar data. While not specifically addressed, one could apply this technique to source encode vehicle state vectors. However, the transform codes provide at best an approximation of the original signal until the entire sequence is received. Furthermore, the performance of the transform compressor improves with longer sequences of data. These limitations make this technique less suitable when near-realtime telemetry of accurate vehicle positions is required, and more suitable for less time-sensitive transmission of previously collected data.

Outside the marine domain resides the closest related work, by Koegel and Mauve [10]. They investigate the information content of a moving urban or highway land vehicle trajectory (defined as a time series of vehicle positions). In this domain, the throughput is much less limited, but the desired number of trajectories to transmit is high. Thus, the ratio of trajectory number to available throughput is similar to the marine domain where we have a small number of trajectories, but a very low throughput link. Koegel and Mauve suggest the use of a Kalman filter for this problem but do not further investigate it, as is done in this paper.

Others have looked at techniques to losslessly encode very large sets of trajectories from terrestrial GPS data, such as the linearization and clustering approach from [11] and the road-network algorithm in [12]. Many of these techniques are focused on the problem of efficiently storing and transmitting full datasets "offline". In the marine domain, it is typically far easier to offload previously collected datasets after vehicle recovery or over electromagnetic wireless links after the vehicle surfaces. Thus, this paper focuses on a technique intended to telemeter realtime or near-realtime data ("online"), which is the more pressing problem for underwater systems due to the highly constrained acoustic link.

II. Approach

A. Application / presentation network layers

The goal of this system is to transmit a sampling (at sample period τ) of a time series of vehicle positions $\mathbf{y}(t)$ where

$$\mathbf{y}(t) = \begin{bmatrix} t \\ x(t) \\ y(t) \\ z(t) \end{bmatrix} \tag{1}$$

is the Cartesian position of the vehicle with reference to a common known datum, with z given as the negative of the vehicle's depth[1]. $\mathbf{y}(t)$ is quantized to a desired quantity (e.g. 1 meter), which is chosen based on the precision needed by the *receiver*. The work presented in this paper can be considered to reside on the application and/or presentation layers in the Open System Interconnection Reference Model [15].

Position measurements are transmitted as one of two types of messages:

- Full transmissions: The vector $\mathbf{y_f}$ which includes the time and full position of the vehicle relative to the experiment datum where

$$\mathbf{y_f} = \begin{bmatrix} t_f \\ x(t_f) \\ y(t_f) \\ z(t_f) \end{bmatrix} \tag{2}$$

  This message is used once at the start of each mission to synchronize the states of the *sender* and *receiver*. The time t_f represents the mission start.
- Delta transmissions:

$$\mathbf{dy}[n] = \begin{bmatrix} dx[n] \\ dy[n] \\ dz[n] \end{bmatrix} \tag{3}$$

  where $n = 0, 1, 2, 3, \ldots$. This delta transmission is sent continuously following a full

[1] For the purpose of this work, the transformation used from geodetic (latitude, longitude) to Cartesian (local) coordinates does not matter. One could use, for example, the Universal Transverse Mercator transformation with the WGS'84 ellipsoid [13] or the North-East-Down transformation of an earth-centered earth-fixed frame; see [14].

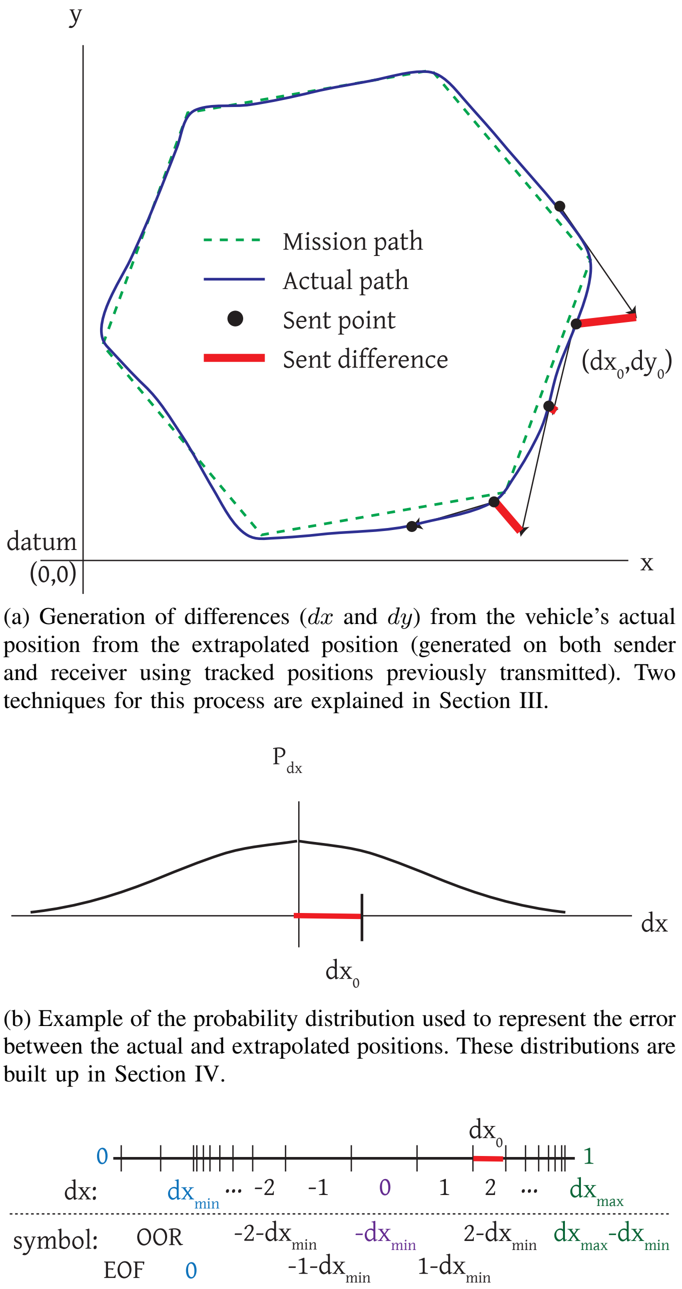
transmission or prior delta transmissions until the vehicle was removed from operation for greater than τ seconds, after which a full transmission is sent to reinitialize the *receiver*'s state. Determining the values of $\mathbf{dy}[n]$ via state observation is described in Section III. The sample number n does not need to be transmitted, assuming the lower layers of the network stack can provide in-order receipt of messages without duplicates. In this case, the decoder simply increments n on each message received. The sampled $\mathbf{y}(t)$ can be reconstructed at time $n = n_0$ using

$$\mathbf{y}(t_f + n_0\tau) = \begin{bmatrix} t_f + n_0\tau \\ \hat{\mathbf{y}}_{prior}[n_0] + \mathbf{dy}[n_0] \end{bmatrix} \tag{4}$$

where $\hat{\mathbf{y}}_{prior}[n] = \mathbf{H}\hat{\mathbf{x}}_{prior}[n]$ is the prior estimate of the state observer extrapolated to discrete time step n. Fig. 1 illustrates the process of generating these delta transmissions and Fig. 2 gives the data flow between the various subsystems presented or used in this paper.

B. Requirements on the lower network layers

As mentioned in section II-A, the requirement that this encoding system has on the lower networking layers is in-order receipt of packets without duplicates. This can be easily accomplished with "Stop-and-Wait" (S&W) Automatic Repeat reQuest (ARQ) along with a single alternating bit to discard duplicates. Specifically, each delta message (or group of messages within a single packet) is transmitted by the *sender* with a field indicating the sequence number of this transmission; using a modulus m of two, this field is a single bit. In this case, each packet is positively acknowledged by the *receiver* before the next packet is transmitted. If one or more acknowledgments is lost, the *sender* retransmits the message (after some timeout). In this case, the *receiver* obtains packets with two identical sequence numbers in a row. The repeated packet is assumed to be a duplicate and is discarded. Such a scheme is suitable for use on an acoustic modem that provides datagram transmission with optional acoustic acknowledgement, such as the WHOI Micro-Modem [17], which was used for the field results in section V-C. This approach is illustrated in Fig. 3, which shows both possible



(a) Generation of differences (dx and dy) from the vehicle's actual position from the extrapolated position (generated on both sender and receiver using tracked positions previously transmitted). Two techniques for this process are explained in Section III.

(b) Example of the probability distribution used to represent the error between the actual and extrapolated positions. These distributions are built up in Section IV.

(c) Arithmetic coding symbol intervals (each dx is mapped to a symbol with 1-meter precision).

Fig. 1: Example illustration for the delta transmissions showing the mapping vehicle position (a) to a given probabilistic model (b) used to generate the symbol intervals required for arithmetic coding (c).

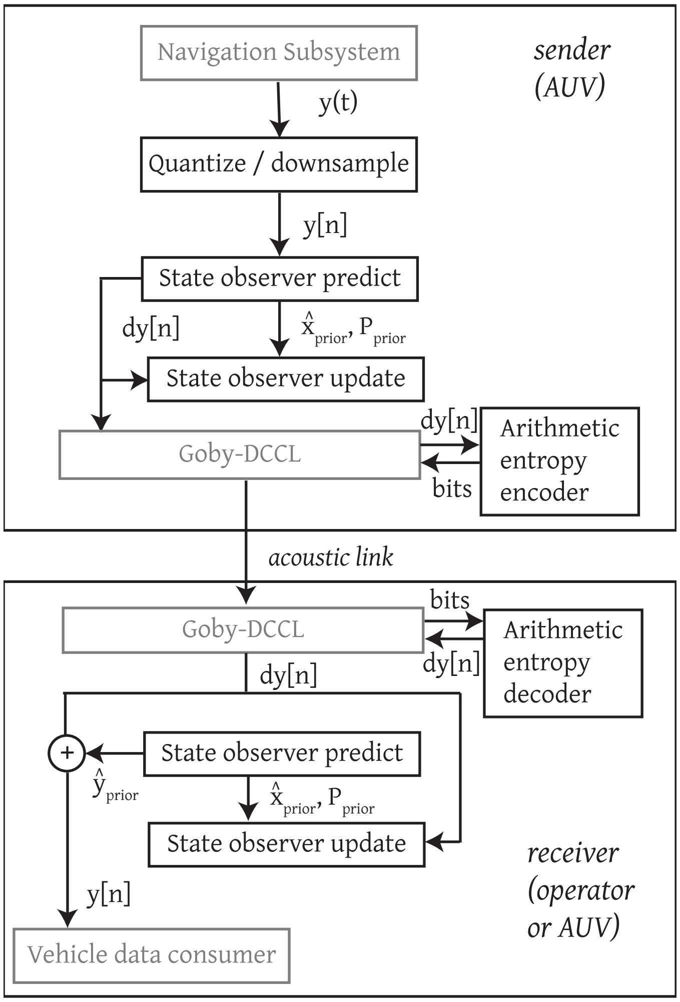


Fig. 2: Data flow for the delta transmission part of the encoding system presented in this paper. Gray blocks are subsystems not presented in this paper. Section III presents the state observers, and section IV elaborates on the arithmetic entropy encoder. Goby-DCCL is discussed in [16].

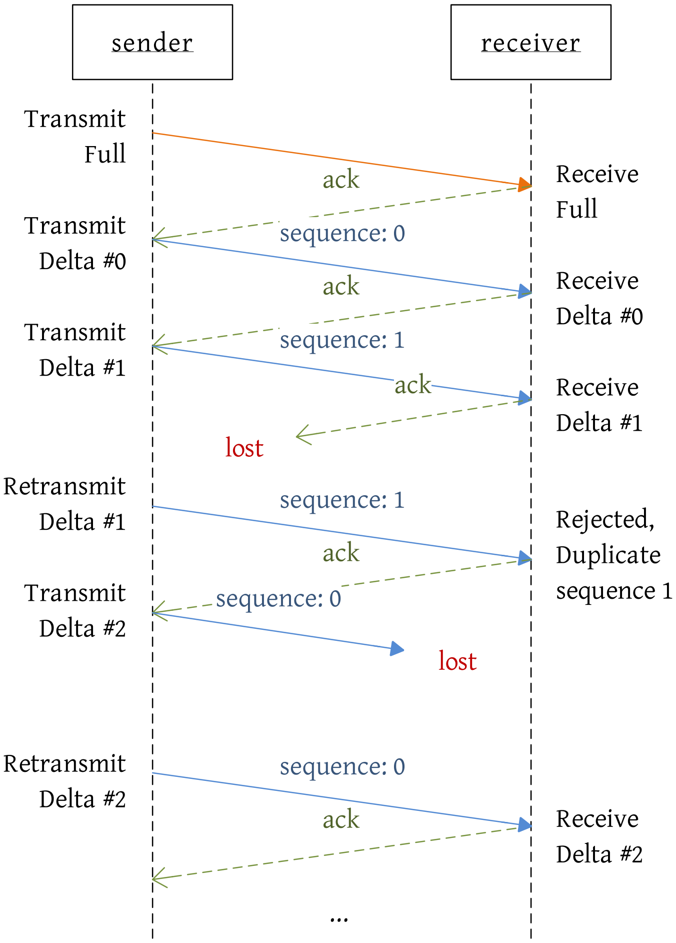


Fig. 3: A sequence diagram showing the full and delta transmissions along with a sequence number (the nth delta message congruent modulo m, with $m = 2$). Both packet loss possibilities are shown: when a packet is lost, it is retransmitted after a timeout (because no acknowledgment is received) and when an acknowledgement is lost, the duplicate is rejected using the sequence number m. The timeout should be chosen using parameters of the physical propagation of the acoustic signal to ensure that the original message was truly lost (and not merely delayed).

failure modes (acknowledgment lost and message lost).

By using a modulus higher than two, it is possible to have more than one packet "in flight" at once, with a single acknowledgment message confirming receipt of all these packets simultaneously. Packets that were missed are selectively retransmitted ("Selective Repeat" ARQ) until all the packets are received. The tradeoff here is fewer acknowledgment messages (and potentially lower latency) at the cost of larger delta messages (due to the larger sequence number field of size $log_2(m)$, in bits). The best choice of ARQ is specific to the goals of the mission and details of the network, and is thus beyond the scope of this paper. For those interested in more detail, Azad et al. [18], explain the aforementioned ARQ strategies and compare their performance in the underwater domain.

III. State Observation

A state observer is typically used in control systems to model the internal state of a system often in order to apply feedback to stabilize the system. Here, a state observer is used in a different way. A model of the system (in this case a vehicle in three-dimensional motion) is observed by both the *sender* and *receiver* of the communications link using a reduced set of the data, namely only the previously telemetered $\mathbf{y}[n]$. The difference between this reduced model and the (presumably more accurate) output of the vehicle's navigation system (which may incorporate other state observers and filters) is taken. This difference (which can be thought of as an error) is the value used to transmit. This operation is visualized in Fig. 1a.

Two state observers were used in this work: a deterministic ("fixed speed") model and a stochastic model based on the Kalman filter.

A. Fixed speed observer

The fixed speed model is useful for AUVs that drive at a roughly constant speed in the xy-plane while underway, which includes most of the torpedo-shaped vehicles such as the Bluefin and REMUS vehicles. This model uses the prior two transmitted positions to determine the predicted direction Ψ of the vehicle's course over ground where

$$\Psi = \tan^{-1} \frac{y[n-1] - y[n-2]}{x[n-1] - x[n-2]} \tag{5}$$

The vehicle's last position is extrapolated using this direction at the fixed speed s, and this is used as a reference for the difference (or error) to the actual vehicle position to be transmitted, such that

$$\begin{bmatrix} dx[n] \\ dy[n] \end{bmatrix} = \begin{bmatrix} x[n] - (x[n-1] + \tau|\mathbf{v}|\cos\Psi) \\ y[n] - (y[n-1] + \tau|\mathbf{v}|\sin\Psi) \end{bmatrix} \tag{6}$$

For depth, since maneuvers are less predictable, the last difference is used:

$$dz[n] = z[n] - (z[n-1] - z[n-2]) \tag{7}$$

The simplicity of this model means that it is computationally inexpensive thus adding negligible overhead to the limited resources on the vehicle. However, it is not applicable for AUVs that can change their speeds substantially while underway. For this, a general purpose model was developed, using the Kalman filter.

B. Kalman filter observer

1) Assumptions: To keep the model as general as possible for a moving vehicle, the following assumptions were made:

- Motion along each Cartesian dimension is independent of the other dimensions.
- The acceleration increment

$$d\ddot{x}[n] = \int_{\tau} \dddot{x}\, dt \tag{8}$$

is a normally distributed white noise process in all dimensions with jerk variance σ_j. In the target tracking literature, this is referred to as the Wiener-sequence acceleration model [19].

These assumptions are somewhat unrealistic (e.g. motion in x and y are rarely independent), but serve to capture the dynamics of the vehicle sufficiently for the given task without introducing significant computational overhead or loss of generality.

2) State space model: Given these assumptions, a linear state space $\mathbf{x}[n]$ is defined as

$$\mathbf{x}[n] = \begin{bmatrix} \mathbf{y}[n] \\ \dot{\mathbf{y}}[n] \\ \ddot{\mathbf{y}}[n] \end{bmatrix} \tag{9}$$

where

$$\mathbf{y}[n] = \begin{bmatrix} x(t_f + n\tau) \\ y(t_f + n\tau) \\ z(t_f + n\tau) \end{bmatrix} \tag{10}$$

The dynamics of the vehicle in discrete time are thus given by

$$\mathbf{x}[n+1] = \mathbf{A}\mathbf{x}[n] + \mathbf{G}d\ddot{x}[n] \tag{11}$$

with state transition model

$$\mathbf{A} = \begin{bmatrix} \mathbf{I_3} & \tau\mathbf{I_3} & \frac{\tau^2}{2}\mathbf{I_3} \\ 0_3 & \mathbf{I_3} & \tau\mathbf{I_3} \\ 0_3 & 0_3 & \mathbf{I_3} \end{bmatrix} \tag{12}$$

where process noise $\mathbf{w}[n]$ is normally distributed

$$\mathbf{w}[n] = \mathbf{G}d\ddot{x}[n] \sim \mathcal{N}(0, \mathbf{Q}) \tag{13}$$

Given the Wiener-sequence acceleration model chosen above, the noise covariance $\mathbf{Q}$ is given as

$$\mathbf{Q} = \sigma_j^2 \mathbf{G}\mathbf{G}^T \tag{14}$$

where

$$\mathbf{G} = \begin{bmatrix} \frac{\tau^2}{2} & \frac{\tau^2}{2} & \frac{\tau^2}{2} & \tau & \tau & \tau & 1 & 1 & 1 \end{bmatrix}^T \tag{15}$$

3) Kalman filter: The Kalman filter [20] is a recursive Bayesian estimator for linear systems with normally distributed noise assumptions. In the marine robotics domain, Kalman filters have been typically used for two purposes: 1) tracking of unknown targets based on noisy and infrequent (often sonar) measurements as in [21], [22]; and 2) estimation of the vehicle's navigation solution from a variety of noisy sensors such as gyroscopes, inertial measurement units, pressure sensors and acoustic sensors (long baseline, Doppler velocity logging, altimeters), such as presented in [23]–[25].

For this work, the Kalman filter is used to predict the state of the system based on a reduced set of measurements $\{\mathbf{y}[n-1], \mathbf{y}[n-2], \ldots\}$, namely those measurements that have already been successfully transmitted to the *receiver*. In a sense, this is similar to the target tracking problem, except that the "target" (the *sender*) is an AUV controlled by the user of the system. This "target" is tracking itself using only the knowledge that the *receiver* (who is also tracking the AUV) has. The goal, as previously mentioned, is to efficiently communicate a more accurate state vector with as few bits as possible. The error between the prediction and the measured state of the system (which is typically a more accurate prediction from the *sender*'s navigation system, which may employ various stochastic filters as well) forms the delta transmission vector $\mathbf{dy}[n]$, also called the "innovation" or measurement residual.

The algorithm presented in this work can be described as a three-step process:

1) Both *sender* and *receiver* predict the next state vector $\hat{\mathbf{x}}_{prior}[n]$ and *a priori* estimate covariance $\mathbf{P}_{prior}[n]$ where

$$\begin{aligned} \hat{\mathbf{x}}_{prior}[n] &= \mathbf{A}\hat{\mathbf{x}}_{post}[n-1] && (16) \\ \mathbf{P}_{prior}[n] &= \mathbf{A}\mathbf{P}_{post}[n-1]\mathbf{A}^T + \mathbf{Q} && (17) \end{aligned}$$

2) The *sender* losslessly encodes (Section IV) and transmits the delta vector

$$\mathbf{dy}[n] = \mathbf{y}[n] - \mathbf{H}\hat{\mathbf{x}}_{prior}[n] \quad (18)$$

using the mapping of estimate to measurement state vectors given by

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}_3 & 0_3 & 0_3 \end{bmatrix} \quad (19)$$

This delta vector is then received and decoded by the *receiver*. At this point, the "true" position of the vehicle can be recovered using (4).

3) Both ends update the filter state vector $\hat{\mathbf{x}}_{post}[n]$ and estimate covariance $\mathbf{P}_{post}[n]$ with their respective posteriors

$$\begin{aligned} \hat{\mathbf{x}}_{post}[n] &= \hat{\mathbf{x}}_{prior}[n] + \mathbf{K}[n]\mathbf{dy}[n] && (20) \\ \mathbf{P}_{post}[n] &= (\mathbf{I} - \mathbf{K}[n]\mathbf{H})\mathbf{P}_{prior}[n] && (21) \end{aligned}$$

using the innovation in combination with the Kalman gain

$$\mathbf{K}[n] = \mathbf{P}_{prior}[n]\mathbf{H}^T\mathbf{S}[n]^{-1} \quad (22)$$

where

$$\mathbf{S}[n] = \mathbf{H}\mathbf{P}_{prior}[n]\mathbf{H}^T + \mathbf{R} \quad (23)$$

The measurement input to the Kalman filter ($\mathbf{y}[n]$) is produced by a process outside of this technique (typically the navigation subsystem of the AUV). Thus, it is possible (or perhaps likely) that the process producing $\mathbf{y}[n]$ is non-Gaussian. In this case, the optimality of the Kalman filter is not guaranteed, and it can diverge completely. Neither of these is detrimental to this system, however. Non-optimality may show up as an increased cost to encode (though not necessarily when using the adaptive distribution given in Section IV), and divergence causes a new full transmission to be sent (when the state observer exceeds $\mathbf{dy}_{min}$ or $\mathbf{dy}_{max}$), thus re-initializing both *sender* and *receiver*.

IV. Arithmetic coding

In the previous section a method was discussed for producing a minimal set of (presumed independent) delta values to transmit. A source encoder can now be chosen to compress these differences.

In this work, arithmetic coding was chosen over various alternatives because of two main advantages:

- Assuming an accurate model, it produces a nearly optimal encoded bitset.
- The modeling process is separate from the coder design. This allows a single implementation of an arithmetic coder to function on many distinct sources of data. It also allows for various models to be evaluated on the source data without redesigning the coder.

The main drawback is that arithmetic coding has a reasonably high computational cost. This is generally not a concern for the underwater vehicle domain since available computing resources typically far outpace the throughput of the acoustic channel.

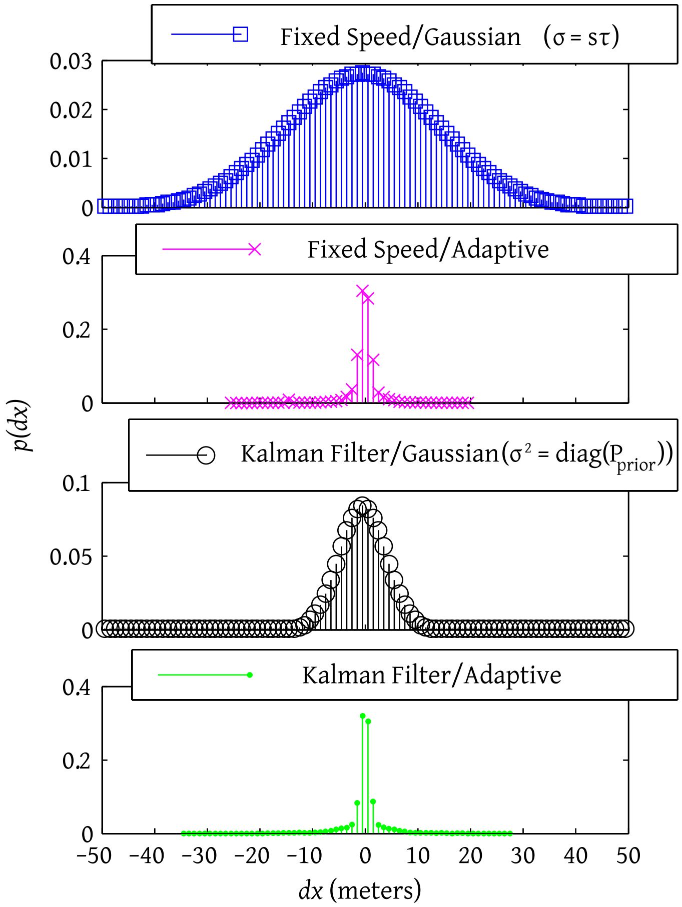


Fig. 4: The probability distributions (given in (26-28)) used to arithmetically encode dx and dy in this work. These models were used to produce the experimental results given in Fig. 7. Not shown is the uniform distribution given in (24).

A. Generating a source model

The next step in this process is identifying a suitable model. The full transmissions (2) are encoded using a uniform probability distribution, since the vehicle could reasonably be redeployed anywhere in the operation region. The process of mapping the source delta data from Section III is sketched in Fig. 1b.

A priori, it seems logical that the probability distribution governing the delta values $\mathbf{dy}[n]$ would be zero mean, since any maneuver that the vehicle performs will have an equal number of negative and positive position differences. For example, see the hexagon in Fig. 1a. The negative dx on the east side will be offset by the positive dx on the west side. The shape of the distribution is unclear, however, and depends substantially on the maneuvering choices the vehicle makes (tight circles would lead to high error using the dynamic model given in (6), straight lines would be low error). Thus, the following distributions were compared (all the non-uniform distributions are shown in Fig. 4):

- Uniform (similar for $p[dy]$ and $p[dz]$):

$$p[dx] = \begin{cases} \frac{1}{dx_{max} - dx_{min} - 1} & dx \in [dx_{min}, dx_{max}) \\ 0 & dx \notin [dx_{min}, dx_{max}) \end{cases} \tag{24}$$

  where the limits

$$\mathbf{dy}_{min} = \begin{bmatrix} dx_{min} \\ dy_{min} \\ dz_{min} \end{bmatrix}, \mathbf{dy}_{max} = \begin{bmatrix} dx_{max} \\ dy_{max} \\ dz_{max} \end{bmatrix} \tag{25}$$

  must be determined *a priori* based on the tolerance for unencodable symbols if the state observer difference exceeds these bounds. In a real system a symbol can be reserved for out-of-range values and the encoder reset to send a new "full transmission" (2) when this occurs. The tighter the bounds, however, the less probability mass that is "wasted" on encoding values that will never or rarely occur.
- Normal, with variance $\sigma^2 = (s\tau)^2$:

$$p[dx] = \begin{cases} \mathcal{N}(0, \sigma^2) & dx \in [dx_{min}, dx_{max}) \\ 0 & dx \notin [dx_{min}, dx_{max}) \end{cases} \tag{26}$$

  The standard deviation $s\tau$ was chosen so that all possible maneuvers including the "worst case" scenario have about 95% of the probability mass. The "worst case" is where the vehicle makes a 180° turn immediately after the preceding transmission so that $dx[n] = 2s\tau$. This means that $\sigma = s\tau$ since

$$\Phi(\mu + 2.0\sigma) - \Phi(\mu - 2.0\sigma) = 0.95 \tag{27}$$

  where Φ is the cumulative mass function of the normal distribution. This distribution is used only in conjunction with the fixed speed observer (Section III-A).
- Normal, with variance $\sigma^2 = diag(\mathbf{P}_{prior}[n])$: that is, the variances of the *a priori* estimate covariance from the Kalman filter. This distribution is used only in conjunction with the Kalman filter observer (Section III-B3).
- Adaptive: This model starts processing the dataset with the uniform distribution given above, and then equally incorporates the statistics of all previously transmitted symbols from

the relative frequencies of the previously encountered values. Thus, an accurate model of the vehicle's prior positions is built up to encode future positions. At any sample m_0, the model is

$$p[dx] = \begin{cases} (f_{dx} + 1) / f_0 & dx \in [dx_{min}, dx_{max}) \\ 0 & dx \notin [dx_{min}, dx_{max}) \end{cases} \tag{28}$$

where f_{dx} is the frequency of the value dx from all the prior transmissions $[dx[1], \ldots, dx[m_0 - 1]]$, except at the start of the experiment ($m = 1$) where $f_{dx} = 0$. f_0 is a normalizing constant given as

$$f_0 = m_0 + dx_{max} - dx_{min} \tag{29}$$

A value of one is added to f_{dx} in (28) so that a given dx can be encoded before it is encountered (otherwise such a value could never be encoded). The model is updated after encoding and after decoding so that the *sender* and the *receiver* can share the same state.

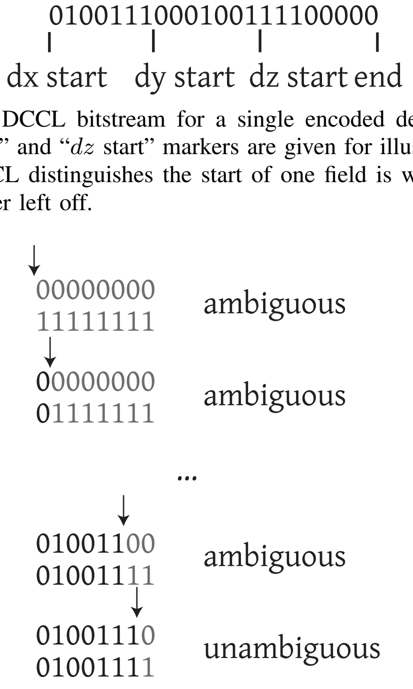
B. Implementing the arithmetic coder

To ensure this work can be easily fielded on AUVs, the arithmetic coder was implemented in the Dynamic Compact Control Language (DCCL), part of the Goby2 project [16]. The details of the arithmetic coder were based on the widely used integer implementation by Witten, Neal, and Clearly [26] and further clarified in [27]. While the integer implementation is used in the code to avoid underflow, overflow, and precision problems, this paper uses the floating point notation for clarity. This notation involves encoding a range from [0, 1) using normalized probability models.

The mapping from delta values to symbol space S (shown in Fig. 1c) is given by

$$S[dx] = \begin{cases} dx - dx_{min} & dx \in [dx_{min}, dx_{max}) \\ \text{out-of-range} & dx \notin [dx_{min}, dx_{max}) \\ \text{end-of-file} & dx \in \emptyset \end{cases} \tag{30}$$

with two special symbols: end-of-file (EOF) used to indicate the end of encoding, and out-of-range (OOR) used to indicate any value outside $[dx_{min}, dx_{max})$. An EOF symbol is not required if the number of messages encoded per packet is arranged between sender and receiver ahead of time.



(a) Example DCCL bitstream for a single encoded delta message. The "dy start" and "dz start" markers are given for illustration only; the way DCCL distinguishes the start of one field is where the last field's decoder left off.

(b) Example decoding dx from this example bitstream. Both the upper and lower bounds are tracked, consuming a single bit at a time until the range unambiguously identifies a symbol.

Fig. 5: Example of the arithmetic decoder for DCCL, showing tracking of decoded ranges to ensure the number of bits consumed by the encoder and the decoder are identical.

The algorithm for arithmetic coding is well known and will thus not be reprinted here for brevity's sake. However, one innovation was required to conform to the DCCL requirement that decoders consume exactly the same number of bits as the encoder produces. The implementation of an arithmetic coder given in [26] and elsewhere assumes that the decoder can safely read nonsense bits past the end of the file, until the actual end-of-file symbol is decoded. This will not work with DCCL since extra bits used in decoding end up being taken from those required for the *next* field in the message and thereby corrupting all following fields. Thus, in the DCCL implementation used here, the decoder tracks both the upper (current bitset followed by all ones) and lower (current bitset followed by all zeros) bounds of the current symbol, adding bits one at a time until the symbol is unambiguously decoded. An example of this process is given in Fig. 5. Relatedly, the end of the bitstream must be encoded exactly so that the decoder does not leave extra bits in the stream that would corrupt the next field of the message. The authors of [26] always use two bits to indicate which middle quarter (either [0.25, 0.5) or [0.5, 0.75)) is wholly contained by the final encoder range. However, when at least one end of the encoder range is at one of the bounds (low = 0 and/or high = 1), fewer bits may be required. The exact set of end bits (e) is given by

$$e = \begin{cases} \emptyset & \text{high } = 1, \text{low } = 0, \text{ no follow bits} \\ 0 \text{ or } 1 & \text{high } = 1, \text{low } = 0, \text{ follow bits} \\ 1 & \text{high } = 1, 0 < \text{low } < 0.5 \\ 0 & 0.5 < \text{high } < 1, \text{low } = 0 \\ 01 & \text{low } < 0.25, \text{high } \geq 0.5 \\ 10 & \text{low } < 0.5, \text{high } \geq 0.75 \end{cases} \tag{31}$$

plus any follow bits accrued from prior center expansions around [0.25, 0.75). This is consistent with the (rounded-up) information entropy

$$\lceil H_{bits} \rceil = -log_2(p) = \begin{cases} 0 & \text{high } = 1, \text{low } = 0 \\ 1 & \text{high } = 1, 0 < \text{low } < 0.5 \\ 1 & 0.5 < \text{high } < 1, \text{low } = 0 \\ 2 & \text{low } < 0.25, \text{high } \geq 0.5 \\ 2 & \text{low } < 0.5, \text{high } \geq 0.75 \end{cases} \tag{32}$$

for the cases in (31).

V. Results on experimental data

Here we will examine the performance of the encoding system developed in the previous sections on transmitting hypothetical messages pulled from two experimental datasets:

- The shallow water GLINT10 experiment in the Tyrrhenian Sea containing in excess of sixty hours of cumulative dive time with a Bluefin 21” AUV.
- A dive from the Arctic Gakkel Vents expedition (AGAVE07) with a SeaBED AUV performing a survey at 4 km depth. The dive was twenty-one hours in duration.

These two datasets were chosen to contrast significantly different AUV classes performing different missions to demonstrate the broad applicability of this approach. Specific quantities from the experiments and values chosen here for these examples are given in Table I. For the results on these datasets, an error free physical link without duplicates is assumed for the purpose of evaluating the performance of the encoding system alone. Section VII explores the performance over a more realistic link with non-zero packet loss.

Finally, this system was implemented and run in the field during the MBAT12 trial using a Bluefin 21” AUV and the WHOI acoustic Micro-Modem.

A. GLINT10

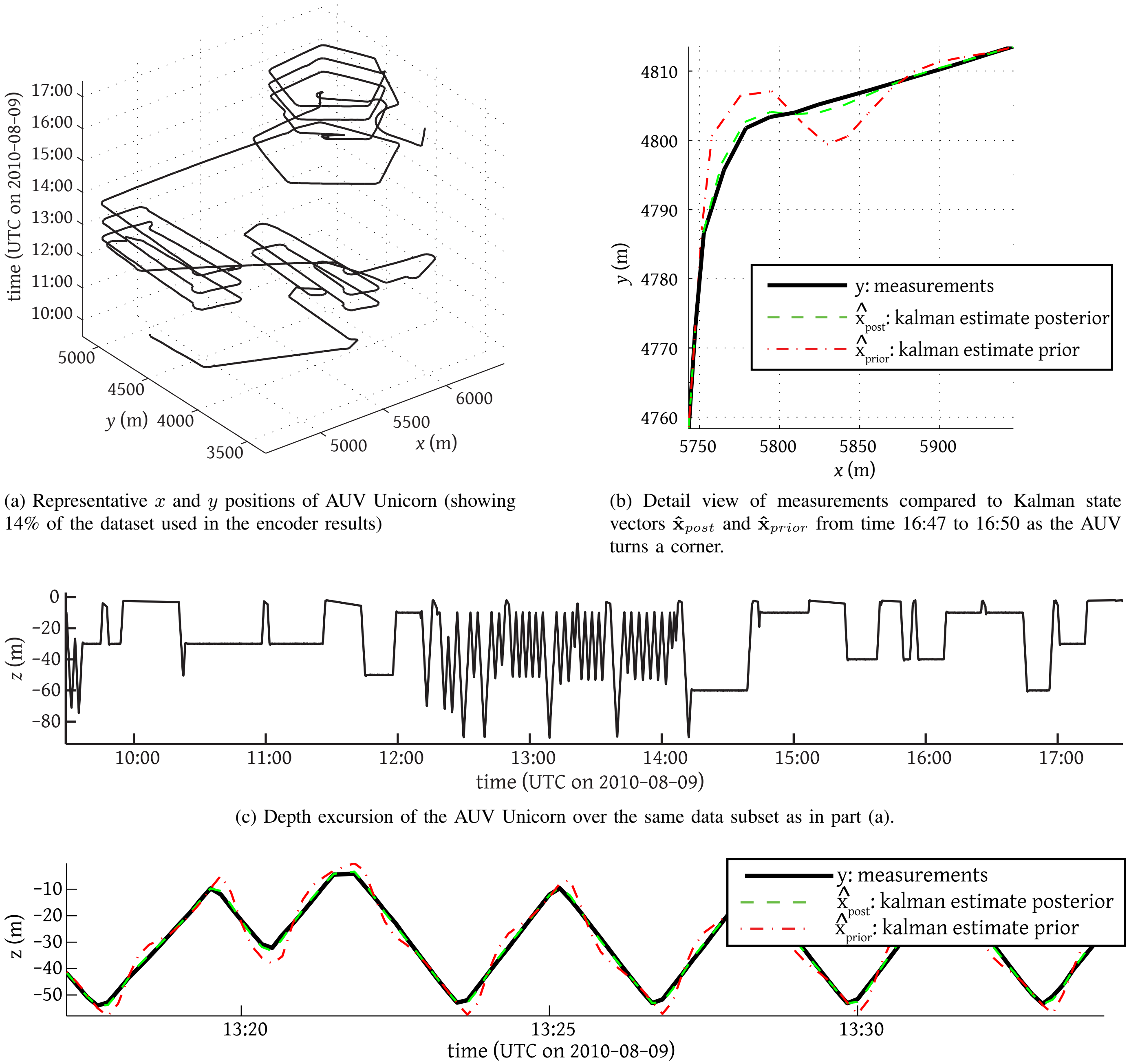
The desired transmission in this example is a Cartesian representation of the vehicle’s position where

$$\begin{bmatrix} x[n] & y[n] \end{bmatrix} = UTM_{WGS84}(lon[n], lat[n]) - UTM_{WGS84}(lon_d \tag{33}$$

and $z[n]$ is the negative of the pressure-derived vehicle depth. UTM_{WGS84} is the Universal Transverse Mercator transformation using the WGS’84 ellipsoid [13], $lon[n], lat[n]$ are the vehicle’s longitude and latitude, and lon_d, lat_d are the longitude and latitude of the experiment datum, a reference used for convenience (unrelated to the UTM zone datum).

1) Position data: A representative subset of the data used is plotted in Fig. 6, and represents one AUV performing a variety of data collection and adaptive autonomy missions. The details of the missions are not of interest here, as the goal is to develop a technique for communicating position data regardless of the vehicle’s mission. As can be seen from Fig. 6a, the AUV performed a variety of polygonal excursions interrupted by straight-line waypoints. In depth, both profiling “yoyo” and constant depth maneuvers were used. In Figs. 6b and 6d, the Kalman filter state vectors are plotted, as well as the measured position of the vehicle from the navigation system. The prior estimate deviates the most when the vehicle maneuvers. This is expected since the causal motion model developed in Section III-B3 has no way to predict these maneuvers and once they occur they are tracked as random changes in the vehicle’s jerk. A more specific motion model would likely improve the tracking here, but at the cost of loss of general applicability to a variety of vehicles and mission types. In any case, the required causality of the model will always limit the performance of this system to some degree.

2) Encoder results: Each of the distributions given in Fig. 4 was used with the arithmetic coder discussed in the prior section to encode the dataset. The resulting size of each message was recorded and the statistics plotted in Fig. 7a, along with the



(a) Representative x and y positions of AUV Unicorn (showing 14% of the dataset used in the encoder results)

(b) Detail view of measurements compared to Kalman state vectors $\hat{\mathbf{x}}_{post}$ and $\hat{\mathbf{x}}_{prior}$ from time 16:47 to 16:50 as the AUV turns a corner.

(c) Depth excursion of the AUV Unicorn over the same data subset as in part (a).

(d) Zoom of part (c) showing the Kalman state vectors in z (negative depth). The value transmitted (dz) is the difference between the measurement and the $\hat{\mathbf{x}}_{prior}$ as given in (18). As shown here, this difference is highest following a sharp manuever.

Fig. 6: Example subset of AUV Unicorn navigation data used for the experimental analysis from the GLINT10 cruise.

TABLE I: Experimental Parameters

Parameter	GLINT10	AGAVE07	MBAT12
Delta model bounds: $[\mathbf{dy}_{min}, \mathbf{dy}_{max})$	[-50, 51) m		
Transmitted x, y, z precision	1 m		
Jerk variance[a] (σ_j^2)	10^{-3}		
Measurement covariance[b] ($\mathbf{R}$)	$25\mathbf{I_3}$		
Time between messages (τ)	10 s	10 s	5 s
Number of full transmissions (N_f)	199	34	1
Mean size of full transmission	60 bits	61 bits	55 bits
Number of delta transmissions (N_d)	24420	7370	660
Size of delta transmission	See Fig. 7 & Table II		
Vehicle xy speed (s)	1.5 m/s	0.2 m/s	1.4 m/s
Water depth (D)	110 m	4140 m	20 m
Experiment datum (lat_d, lon_d)	42.45667°N, 10.875°E	85.61667°N, 85.75°E	42.38°N, 70.96°W

[a] The jerk variance was determined using a subset of the GLINT10 data. See section VI for further discussion.

[b] This measurement error is chosen as a conservative value (2σ = 10 meters) for the random noise in the $\mathbf{y}(t)$ values from the vehicle's navigation solution. Lower values may improve performance (but at the risk of loss-of-lock of the Kalman filter).

uncompressed 32-bit integer as a reference point. The moments of these results are summarized in Table II. As expected, the Gaussian model performed better than the uniform distribution since it makes use of the dynamic models from Section III where low errors are more probable than high errors (the vehicle in general continues on a similar path of motion). However, both of the Gaussian distributions overstate the error significantly from the adaptive distribution, as seen by the difference in standard deviation between the two models in Fig. 4. Once the adaptive distribution was initialized with sufficient data, it easily outperforms the results using the other distributions. Furthermore, the fixed speed dynamic model using the adaptive distribution is an improvement of 86% over the widely used Compact Control Language [28], which uses 61 bits to encode a vehicle position in the "MDAT_STATE" message.

Comparing the two dynamic models used, the Kalman filter has a significant edge with the Gaussian distribution since it produces an uncertainty model ($\mathbf{P}_{prior}$) as part of the state estimation process. For the adaptive distribution, however, the fixed speed model performs slightly better, especially since it has lower standard deviation due to a smaller number of large (i.e. 16-20 bit) messages.

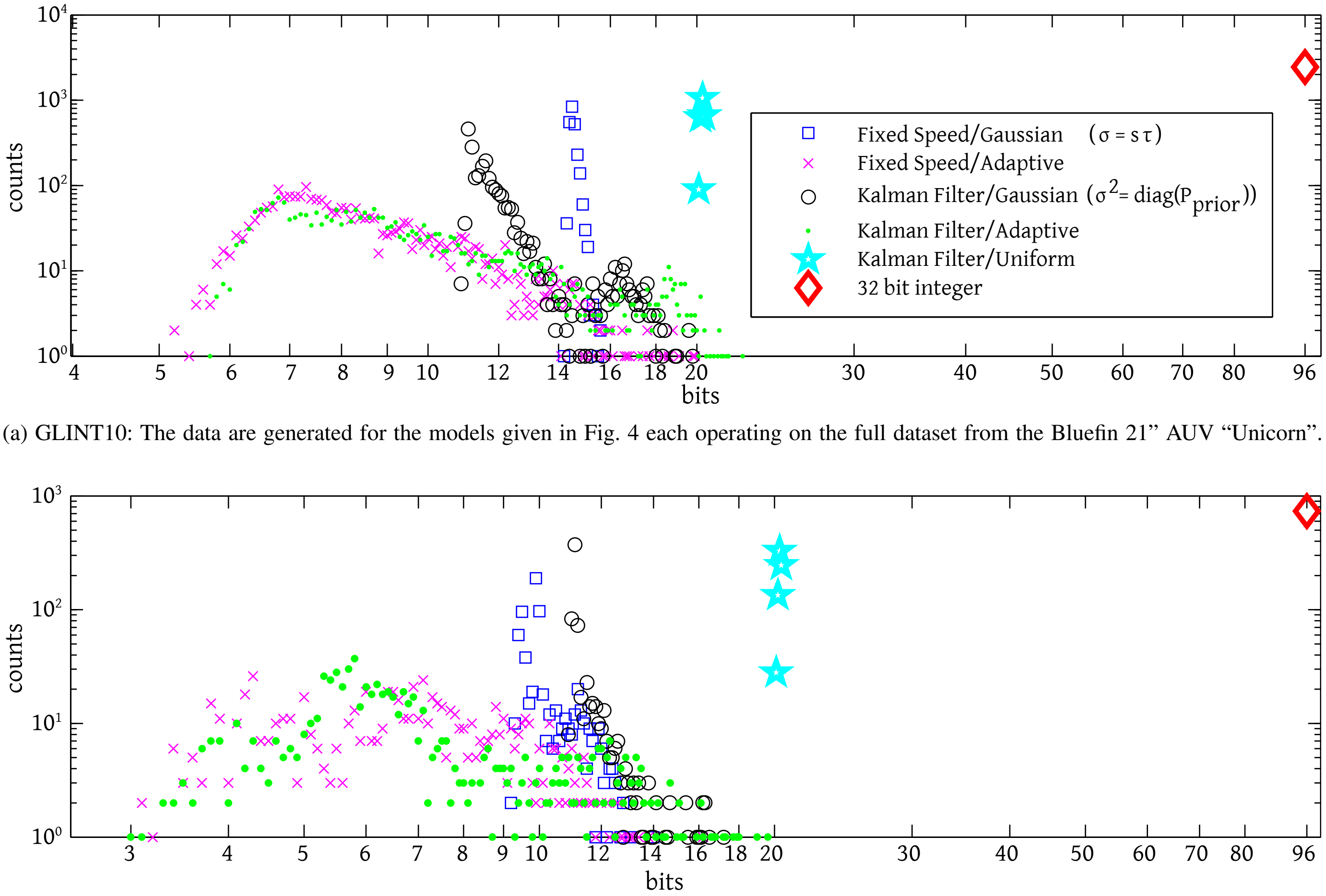
B. AGAVE07

As with the GLINT10 dataset, the vehicle's Cartesian position during AGAVE07 (Fig. 8) was hypothetically transmitted, but this time using the AlvinXY transformation from latitude and longitude to x and y [14]. The size of each delta message was computed for the same types of models as for the GLINT10 experiment; these results are plotted in Fig. 7b. In general, the results from the two datasets are similar. There are two main differences: 1) the overall size of the messages generated from the non-uniform distributions are smaller for the AGAVE07 dive than for the GLINT10 trial; and 2) the fixed speed model outperforms the Kalman filter model on the Gaussian distribution for the the AGAVE07 dataset.

Both differences are likely due to the difference in vehicle speeds. Since the resolution transmitted (1 m) and time step ($\tau = 10$ s) were kept constant between experiment datasets, the slower vehicle ("Jaguar" from AGAVE07) will diverge less from the expected position. The variance of the normal distribution for the fixed speed model is based on speed. In the case of the AGAVE07 results, this is a narrower distribution, more closely matching the adaptive model than in the GLINT10 case.

C. MBAT12

In this experiment, the technique described in this paper was demonstrated *in situ* on-board a Bluefin 21" AUV (the *sender*) equipped with a WHOI acoustic Micro-Modem [17]. The *receiver* was a buoy equipped with both a Micro-Modem and a radio link to the research vessel. For comparison, the state of the vehicle was transmitted at each



(a) GLINT10: The data are generated for the models given in Fig. 4 each operating on the full dataset from the Bluefin 21" AUV "Unicorn".

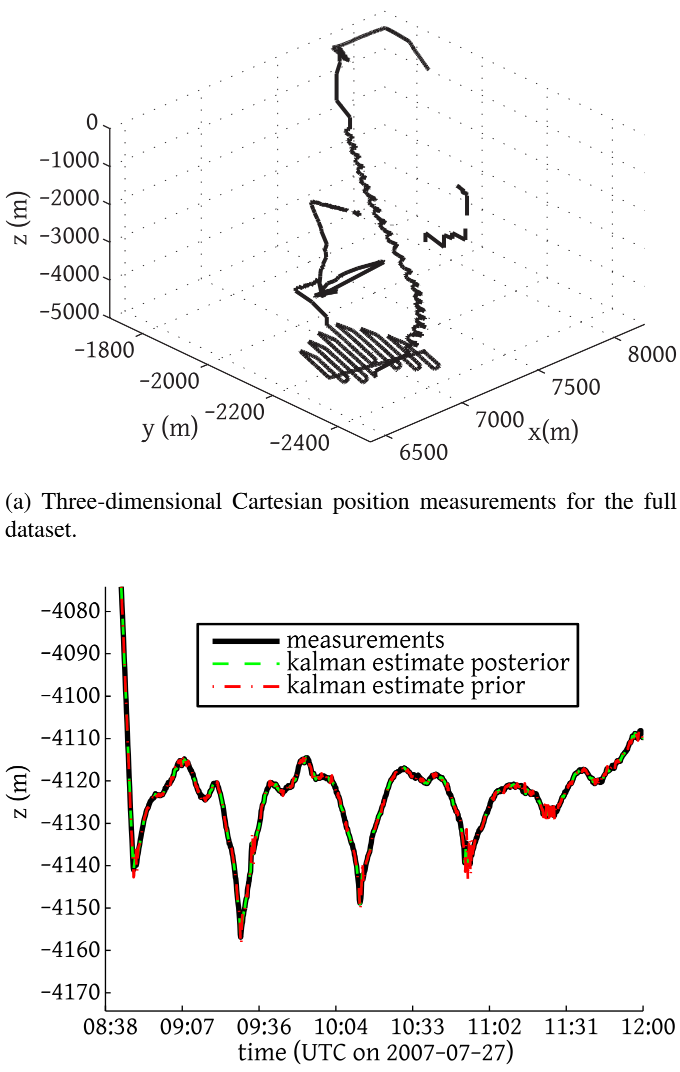
(b) AGAVE07: Results from the SeaBED AUV "Jaguar" dive on 2007-07-27 using the same distribution types.

Fig. 7: Log-log plot of the number of delta messages generated with a given size (in bits) for each experiment. An uncompressed 32-bit integer representation is provided for comparison.

TABLE II: Experimental Results

Dynamic Model	Distribution	Mean[a]	Standard deviation[a]	Compression [b]
		GLINT10		
Fixed speed	Gaussian ($\sigma = s\tau$)	14.6	0.157	85%
Fixed speed	Adaptive	8.68	2.22	91%
Kalman filter	Gaussian ($\sigma^2 = diag(\mathbf{P}_{prior}[n])$)	12.0	1.42	87%
Kalman filter	Adaptive	9.76	3.28	90%
Kalman filter	Uniform	20.3	0.0821	79%
		AGAVE07		
Fixed speed	Gaussian ($\sigma = s\tau$)	10.1	0.746	89%
Fixed speed	Adaptive	7.11	2.23	93%
Kalman filter	Gaussian ($\sigma^2 = diag(\mathbf{P}_{prior}[n])$)	11.4	0.839	88%
Kalman filter	Adaptive	7.45	3.00	92%
Kalman filter	Uniform	20.3	0.0814	79%
		MBAT12		
Kalman filter	Gaussian ($\sigma^2 = diag(\mathbf{P}_{prior}[n])$)	11.1	0.481	88%

[a] Mean and standard deviation given in bits.
[b] Relative to a 3 element 32-bit integer representation.



(a) Three-dimensional Cartesian position measurements for the full dataset.

(b) Subset of depth showing the Kalman state vectors.

Fig. 8: SeaBED "Jaguar" dive used from the AGAVE07 experiment for the results in Fig. 7b.

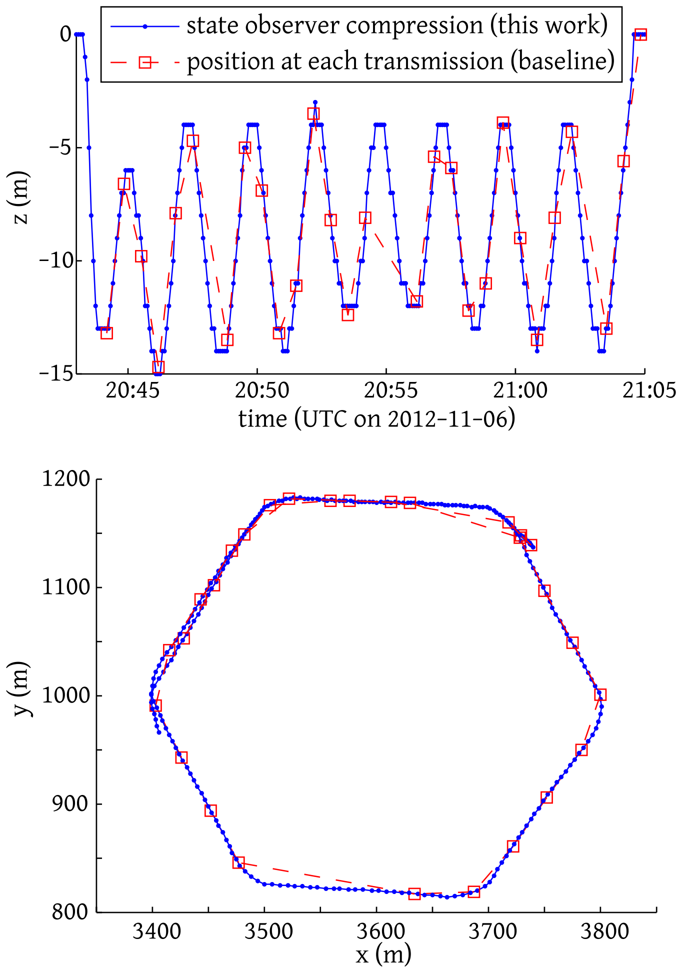
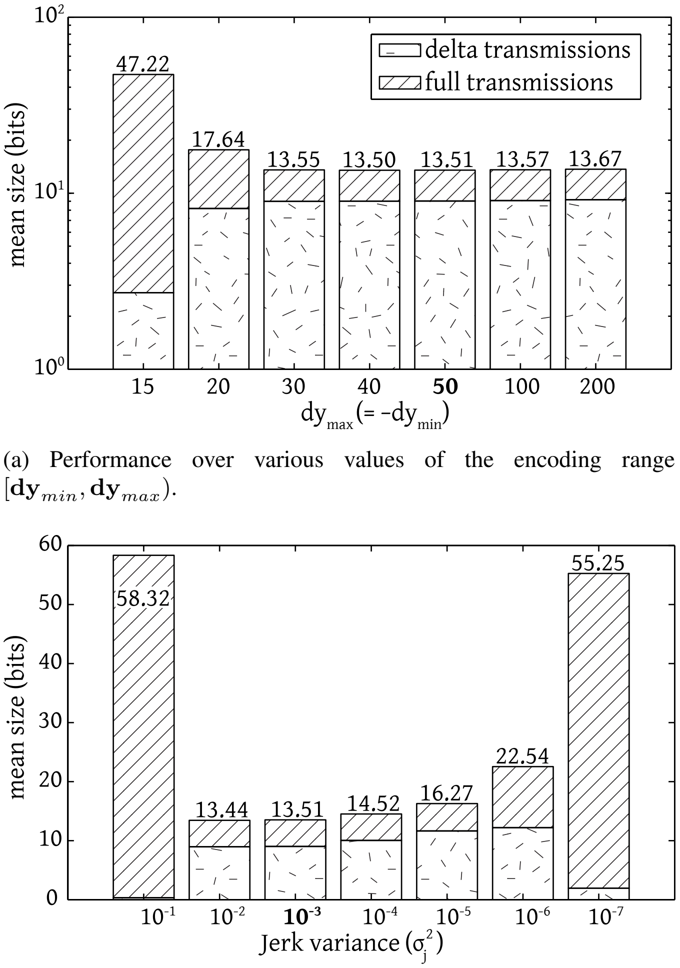


Fig. 9: Position of the *sender* vehicle as seen by the *receiver* during the MBAT12 field trial. The state observer compression technique developed in this paper is compared to the traditional position sent with each acoustic data transmission. Note that this new technique removes the aliasing present in the vehicle's depth excursions.

acoustic datagram transmission in addition to the state estimate innovations (with $\tau = 5$s). Fig. 9 shows the results from a mission during this experiment, which used the Kalman filter state observer coupled with the Gaussion distribution for encoding the delta transmissions. Using the lowest data rate available with the phase-shift-keying (PSK) modulation on the Micro-Modem ("rate 1"), only 12.6% of the available data throughput during this mission was used to transmit the state observer innovations (including addressing, duplicate rejection, and byte padding overhead). Thus, using this technique, it is possible to consider telemetry of vehicle positions as a reasonably small amount of overhead on the underwater communications system, rather than its primary purpose as has historically been the case.

VI. Robustness

The design of the state observers is intentionally general to reduce the number of parameters to be "tweaked" or "tuned" and thus improve the robustness (for a broad range of maneuvering vehicles). However, there are two major parameters to determine that have values which are not clear *a priori*: the range of included delta values for the arithmetic entropy encoder (i.e. $[\mathbf{dy}_{min}, \mathbf{dy}_{max})$) and, when using the Kalman filter estimator, the process noise (embodied in its variance: σ_j^2) which is used to model all vehicle maneuvers.



(a) Performance over various values of the encoding range $[\mathbf{dy}_{min}, \mathbf{dy}_{max})$.

(b) Performance dependence on the variance of the process noise (which is how vehicle maneuvers are modeled)

Fig. 10: Using the GLINT10 dataset, the performance (in mean size of messages) of this system over a range of parameter values. The contributions to the mean size from the delta and full transmission are shown separately. The values in bold were used for the rest of the analysis in this chapter and are also given in Table I.

The first parameter ($[\mathbf{dy}_{min}, \mathbf{dy}_{max})$) is illustrated in Fig. 10a, which shows the tradeoff in choosing these bounds. Too tight, and the state observer consistently exceeds the bounds, and a full transmission must be resent to reinitialize both *sender* and *receiver* states. However, there is also a small price to pay for making them too loose, which is the small amount of probability mass required for each discrete value with the range of $\mathbf{dy}_{min}$ to $\mathbf{dy}_{max}$, taking away mass from all the remaining values, making the more probable values (e.g. 1, 0, -1) slightly more costly to encode. However, this cost is small compared to having to reinitialize the states by sending full transmissions frequently. Thus, as is clear from Fig. 10a, it is preferable to err on the side of too loose bounds than too tight.

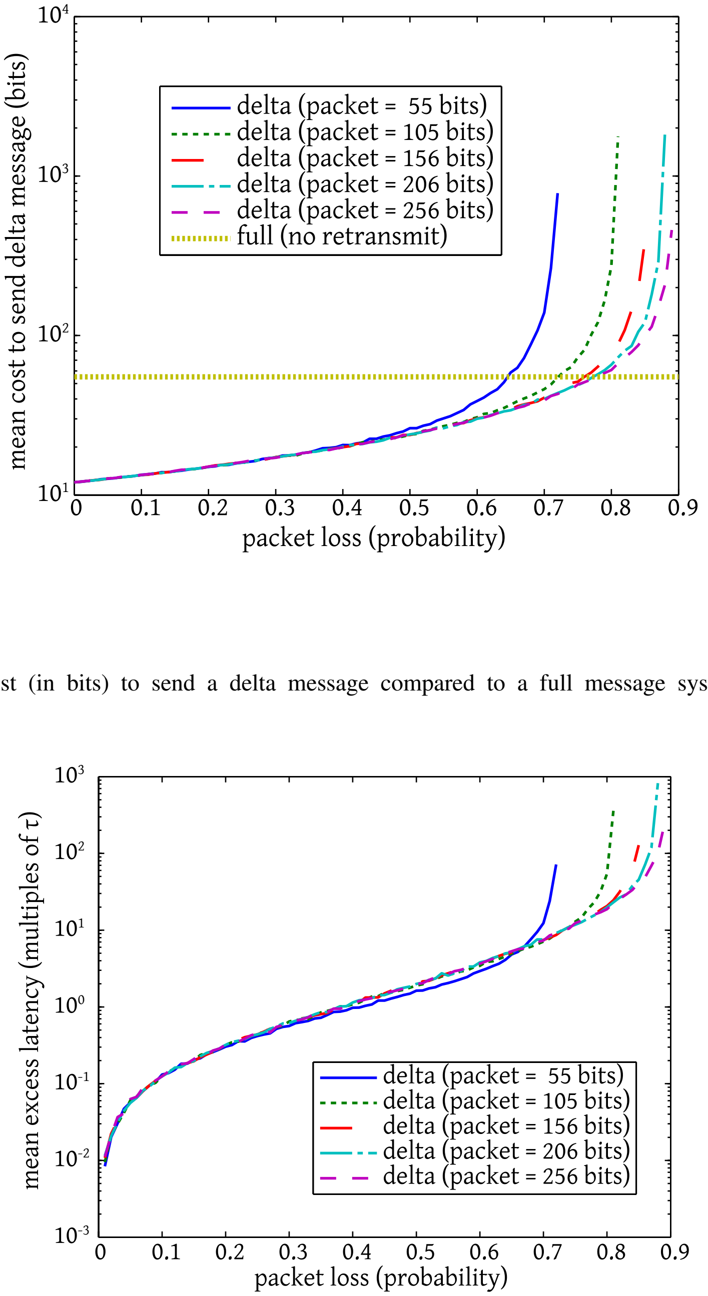
The second parameter (σ_j^2) was originally chosen using a subset of the experimental data from GLINT10 to determine a reasonable order-of-magnitude value ($\sigma_j^2 = 10^{-3}$). Figure 10b shows the overall performance (message size in bits, including full transmissions and delta transmissions) for a wide range of process noise values. This figure shows that the algorithm is robust over about four order of magnitude from 10^{-2} to 10^{-5} (too low or too high and the filter perpetually fails to track). It is also worth remembering that the fixed speed tracker does not require this parameterization and is applicable to a large number of classes of AUVs found in the field today.

Finally, by design, the tradeoff for inaccuracy in determining these parameters is *not* accuracy in the received telemetry, but rather cost (in bits) of sending these data. This means that the *receiver* is never uncertain about the quality of the data that it has received.

VII. Performance comparison to traditional approach

As mentioned in Section II, the delta technique developed in this chapter provides an evenly sampled historical time series (regardless of packet loss) by retransmitting dropped packets (using ARQ). Such an evenly sampled history is useful for in-situ analysis of instrument data attached to these position messages, or performance evaluation. Since ship time is highly expensive, it is valuable to do as much data analysis and debugging as possible while the vehicle is underway (rather than waiting until the end of the mission).

However, there are times when only the latest position of the vehicle is desired. Traditionally, the latest status message is sent in full without the expectation of acknowledgment. However, does this state observation technique still provide any benefit in this case? Using the experimental results from MBAT12 (Table II), the simulated mean cost to send a delta message was plotted against the packet loss (modeled as an independent Bernoulli process) as Fig. 11a. This metric was computed for a variety of packet sizes, ranging from the full transmission



(a) Mean cost (in bits) to send a delta message compared to a full message system with no retransmits.

(b) Increased latency (over full system with no transmits) for a variety of packet sizes using the ARQ scheme illustrated in Fig. 3.

Fig. 11: Performance comparison of traditional “newest when possible” system (full transmission with no retransmits) with the delta state observer technique *in the case when a full historical time series is unneeded.* Where the curves asymptotically approach infinite cost or latency is the point where the channel is too lossy to transmit all the delta messages. In this case, the desired τ would have to be increased.

size (55 bits) to 32 bytes (256 bits), a commonly used acoustic modem maximum transmission unit (MTU). In addition, a curve representing the traditional system using full transmissions with no retransmission upon loss is provided. Thus, depending on the packet size, this delta-based technique is cheaper (uses fewer bits on average) than a traditional full position system up to packet losses of 65% or higher.

On the other hand, the delta system presented in this paper requires that older innovations be correctly received before newer innovations can be used (otherwise the *sender* and *receiver* state observers do not share the same state). This leads to increased latency over the full position system that increases monotonically with packet loss. This excess latency is shown in Fig. 11b, and as would be expected, increases monotonically with increasing packet loss. The tolerance for excess latency versus increased message cost (in bits) is mission specific. A hybrid system that switches between this delta technique (for lower packet loss links) and a traditional full system (for higher packet loss links) may be necessary to reach the desired tradeoff of latency versus throughput for a given mission.

VIII. Conclusion

A technique for transmitting the position vector of an AUV at relatively high rates at very low cost (in bits) was developed and demonstrated on two experimental data sets and implemented in the field, leading to mean compression ratios as high as 93%. While developed and shown here specifically for sending vehicle position vectors, this technique could be extended to send scalar environmental measurements using sensor data coupled with a static or dynamic ocean model. Also, it could be used to transmit the position of an unknown target by using the target's estimated position (computed using an on-board sonar and signal processing subsystem) paired with a dynamic model of the target's motion.

In the case of lossy networks, this technique still provides a reduced message size, even when only the latest position of the vehicle is desired. However, this system does introduce extra latency due to the need to retransmit old messages before sending new ones. Using the results from MBAT12, we showed that this latency is reasonable (a factor of the sampling period τ or less) up to 40% packet loss. Such a system is suitable for use with the currently available acoustic modems that provide only on the order of 10^1 to 10^2 bytes per minute of throughput. This paper shows that continuous telemetry of vehicle position is possible while staying well within the abilities of these modems, allowing for additional mission-specific messaging to take place as well.

Acknowledgment

The authors extend many thanks to Hanumant Singh and the SeaBED group at the Woods Hole Oceanographic Institution for making the AGAVE07 dataset available for use in this work. We also equally appreciate the support of the NATO Centre for Maritime Research and Experimentation (formerly NURC) for making the GLINT10 experiment possible.

References

[1] Y. Zhang, M. Godin, J. Bellingham, and J. Ryan, "Using an autonomous underwater vehicle to track a coastal upwelling front," *IEEE Journal of Oceanic Engineering*, vol. 37, no. 3, pp. 338 –347, Jul. 2012.

[2] S. Petillo, A. Balasuriya, and H. Schmidt, "Autonomous adaptive environmental assessment and feature tracking via autonomous underwater vehicles," in *OCEANS 2010 IEEE - Sydney*, May 2010, pp. 1 –9.

[3] M. Purcell, D. Gallo, G. Packard, M. Dennett, M. Rothenbeck, A. Sherrell, and S. Pascaud, "Use of REMUS 6000 AUVs in the search for the air france flight 447," in *OCEANS 2011*, Sep. 2011, pp. 1 –7.

[4] A. Baggeroer, "An overview of acoustic communications from 2000-2012," in *Underwater Communications: Channel Modelling & Validation*, 2012.

[5] M. Chitre, personal communication, UComms 2012 conference, 2012.

[6] M. Chitre, S. Shahabudeen, and M. Stojanovic, "Underwater acoustic communications and networking: Recent advances and future challenges," *Journal of the Marine Technology Society*, vol. 42, no. 1, pp. 103–116, 2008.

[7] I. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, p. 257279, 2005.

[8] C. Murphy and H. Singh, "Human-guided autonomy for acoustically tethered underwater vehicles," in *OCEANS 2008*, Sep. 2008, pp. 1 –8.

[9] C. Murphy, "Progressively communicating rich telemetry from autonomous underwater vehicles via relays," Ph.D. dissertation, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, 2012.

[10] M. Koegel and M. Mauve, "On the spatio-temporal information content and arithmetic coding of discrete trajectories," *Mobile and Ubiquitous Systems: Computing, Networking, and Services*, p. 1317, 2012. [Online]. Available: http://www.springerlink.com/index/P1032Q10R1542638.pdf

[11] D. Feldman, C. Sung, and D. Rus, "The single pixel GPS: learning big data signals from tiny coresets," in *Proc. 20th ACM International Conference on Advances in Geographic Information Systems*, 2012.

[12] A. Civilis, C. Jensen, and S. Pakalnis, "Techniques for efficient road-network-based tracking of moving objects," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, no. 5, p. 698712, 2005.

[13] NIMA, "Department of defense world geodetic system 1984: Its definition and relationships with local geodetic systems. second edition, amendment 1," NIMA, Tech. Rep. TR8350.2, 2000, available at http://earth-info.nga.mil/GandG/publications/tr8350.2/wgs84fin.pdf, accessed January 3, 2010.

[14] C. Murphy and H. Singh, "Rectilinear coordinate frames for deep sea navigation," in *Autonomous Underwater Vehicles (AUV), 2010 IEEE/OES*. IEEE, 2010, pp. 1–10.

[15] H. Zimmermann, "OSI reference model–The ISO model of architecture for open systems interconnection," *Communications, IEEE Transactions on*, vol. 28, no. 4, pp. 425–432, 2002.

[16] T. Schneider and H. Schmidt, "Goby-acomms version 2: extensible marshalling, queuing, and link layer interfacing for acoustic telemetry," in *9th IFAC Conference on Manoeuvring and Control of Marine Craft, Arenzano, Italy*, 2012.

[17] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball, "The WHOI Micro-Modem: an acoustic communications and navigation system for multiple platforms," in *IEEE Oceans Conference*, 2005.

[18] S. Azad, P. Casari, F. Guerra, and M. Zorzi, "On ARQ strategies over random access protocols in underwater acoustic networks," in *OCEANS, 2011 IEEE-Spain*. IEEE, 2011, pp. 1–7.

[19] X. Rong Li and V. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333 – 1364, Oct. 2003.

[20] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[21] S. Rao, "Modified gain extended kalman filter with application to bearings-only passive manoeuvring target tracking," in *Radar, Sonar and Navigation, IEE Proceedings-*, vol. 152, 2005, p. 239244.

[22] R. Lum and H. Schmidt, "Exploiting adaptive processing and mobility for multistatic tracking by AUV networks," in *Proceedings of 4th International Conference on Underwater Acoustic Measurements: Technologies and Results*, Kos, Greece, Jun. 2011.

[23] M. Blain, S. Lemieux, and R. Houde, "Implementation of a ROV navigation system using acoustic/Doppler sensors and kalman filtering," in *OCEANS 2003. Proceedings*, vol. 3, 2003, p. 12551260.

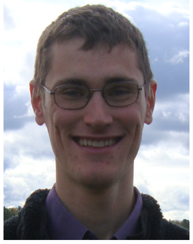
[24] D. Loebis, R. Sutton, J. Chudley, and W. Naeem, "Adaptive tuning of a kalman filter via fuzzy logic for an intelligent AUV navigation system," *Control Engineering Practice*, vol. 12, no. 12, pp. 1531–1539, Dec. 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0967066103002582

[25] S. E. Webster, R. M. Eustice, H. Singh, and L. L. Whitcomb, "Advances in single-beacon one-way-travel-time acoustic navigation for underwater vehicles," *International Journal of Robotics Research*, vol. 31, no. 8, p. 935950, Jul. 2012.

[26] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, no. 6, p. 520170, 1987. [Online]. Available: http://dl.acm.org/citation.cfm?id=214771

[27] K. Sayood, *Introduction to Data Compression*. Elsevier, Dec. 2005.

[28] R. P. Stokey, L. E. Freitag, and M. D. Grund, "A compact control language for AUV acoustic communication," *Oceans 2005-Europe*, vol. 2, p. 11331737, 2005.



Toby Schneider (M'05) received a B.A. in physics at Williams College in Williamstown, MA, USA in 2007, and a Ph.D. in oceanographic engineering in 2013 from the Joint Program between the Massachusetts Institute of Technology (MIT) in Cambridge, MA, USA and the Woods Hole Oceanographic Institution (WHOI) in Woods Hole, MA, USA.

He is currently a postdoctoral associate for the Laboratory for Autonomous Marine Sensing Systems at MIT working on problems at the interface of marine robotic autonomy and communications. Further professional details about Dr. Schneider are available on his website: http://gobysoft.org.

Henrik Schmidt received the M.S. and Ph.D. degrees from the Department of Structural Engineering, Technical University of Denmark, Lyngby, Denmark, in 1974 and 1978, respectively. He is a Professor of Mechanical & Ocean Engineering at the Massachusetts Institute of Technology (MIT), Cambridge. Following a postdoctoral fellowship at the Risoe National Laboratory in Denmark, he joined the Centre for Maritime Research and Experimentation in Italy in 1982, where he worked until he joined the MIT faculty in 1987. His research has focused on underwater acoustic propagation and signal processing, and most recently on the development of environmentally adaptive acoustic sensing concepts for networks of autonomous underwater vehicles.

Prof. Schmidt is a Fellow of the Acoustical Society of America, and he is the 2005 recipient of the ASA Pioneers of Underwater Acoustics Medal.