CoExploration for Adaptive AUV Survey

Laura Lindzey
University of Washington
Applied Physics Laboratory
Ocean Engineering Department
Seattle, Washington
lindzey@uw.edu

Isaac Vandor
Woods Hole Oceanographic Institution
Applied Ocean Physics and Engineering
Woods Hole, MA
ivandor@whoi.edu

Toby Schneider
GobySoft, LLC
Mashpee, MA, USA
toby@gobysoft.org

Carl Kaiser
Saildrone Inc
Fall River, MA
carl.kaiser@saildrone.com

Michael Jakuba
Woods Hole Oceanographic Institution
Applied Ocean Physics and Engineering
Woods Hole, MA
mjakuba@whoi.edu

***Abstract*—Scientific seabed surveys often require the use of multiple sensing modalities with different capabilities and operational requirements. When using AUVs, this is often accomplished via a series of dives, between which operators examine collected data and plan the subsequent survey. Planning a follow-up survey while the vehicle is still in the water dramatically improves operational efficiency, but requires that topside scientists receive information from the initial survey during the dive. With this motivation, we developed a toolbox for CoExploration that is designed to acoustically transmit scientifically-actionable data, making use of any bandwidth that is not required for safe vehicle operation. This paper describes utilities for incrementally transmitting a multi-resolution multibeam map and for progressive transmission of camera imagery, along with field results from their first use on the NUI hybrid AUV/ROV.**

I. Introduction

The benthic deep-sea, with depths ranging from 200 m to 11 km, is the largest habitat on earth (covering about 65% of earth's surface) and it remains largely unexplored. Currently, nearly all benthic exploration is conducted using large ocean-going research vessels that may deploy various subsea assets including Autonomous Underwater Vehicles (AUVs), Remotely Operated Vehicles (ROVs), submersibles, and a wide variety of other towed or lowered sampling and observation equipment. Mapping from ships has resulted in modern bathymetric maps covering about 20% of the seafloor. However, "mapped" is not a synonym for "explored." A more synoptic exploration of the ocean requires data that can only be generated by in situ assets: water column chemisty, imagery, video, physical samples, and high resolution (meter-scale) bathymetric, sidescan and other types of seafloor maps. These have been collected over a vanishingly small portion of the seafloor.

In this work, we pursue an approach to deep-sea exploration called CoExploration. CoExploration is the blending of a human explorer with a robotic explorer into a tightly coupled system that achieves more than either could alone. In ocean exploration in particular, many missions are focused on locating and surveying sparse features of limited size such as methane seeps, coral mounts, and archaeological artifacts. In such missions, two to three entirely separate sensor configurations and survey patterns are needed to go from broad area search to full characterization. In practice, each dive typically follows a pre-planned trajectory and data products become available only post-recovery, so following up on interesting observations in one dive requires at least one additional dive evolution. The CoExploration approach relies on selectively telemetered data, either requested by operators or identified as important by the robot, to enable on-the-fly mission modification informed by both real time data access on the part of the robot, and scientific expertise and contextual awareness on the part of the operators.

This work was funded by the NOAA office of Ocean Exploration and Research under grant NA17OAR0110207 with additional support from the NOAA Ocean Exploration Cooperative Institute (NA19OAR4320072).

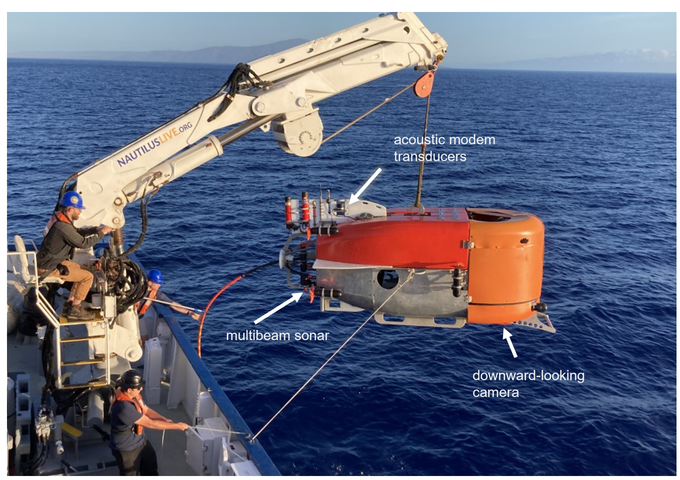

Fig. 1. NUI being recovered after a dive. Annotations show the location of relevant hardware for this work: the 10 kHz and 3.5 kHz transducers, a downward-looking camera, and the Norbit multibeam sonar.

Here we describe a set of CoExploration tools that we developed to enable human operators to better utilize the acoustic modem and the data already collected in order to reduce or eliminate the need for multiple dives. These tools were demonstrated on the NUI hybrid AUV/ROV (Figure 1) during a 2022 cruise aboard R/V Nautilus. Related work on low-level raw data compression and higher-level onboard autonomous data interpretation is highly relevant and briefly summarized subsequently. However, our focus here is on the effective use of the acoustic link in the context of multiple data streams from typical survey modalities (time series data, bathymetry, and photo imagery). Rather than developing a general data compression approach, and treating all data as equivalent byte streams, we considered how a human might want to interact with the data, identifying regions of interest from a lower-resolution representation. For each supported data type, we developed both a scheme for transferring it and a GUI for the scientist to optionally direct what portions of the data should be prioritized.

II. Background & Related work

In a ∼24 hr dive, modern AUVs can generate bathymetric maps covering a few 10s of km^2 at meter-scale resolution (e.g., [Yoerger et al., 2007], [Vaughn et al., 2018], [Kaiser et al., 2016]). Water column chemistry and seafloor imagery may also be collected, though the latter results in a survey footprint orders of magnitude smaller than typical sonar-based mapping products. At present, it is difficult to locate interesting and important features in the data collected by AUVs while they are in the water, e.g., to target photographs over features identified in sonar. Instead, additional, iterative dives (and additional expensive ship time) are required to thoroughly explore a location with multiple sensing modalities. This "nested survey" methodology has proved highly effective in past expeditions. AUV-based exploration for hydrothermal venting employed a nested survey approach consisting of a high-altitude dive focused on the mapping of chemical anomalies in the water column, followed by a bathymetric survey at lower altitude and finally a photographic survey near the seafloor [German et al., 2008]. Each phase required topside processing of the data between dives to target the subsequent phase. Similar approaches have been applied successfully to methane seeps [Brothers et al., 2013], coral "hard-grounds" [Fisher et al., 2014] and in search, e.g., [Purcell et al., 2011]. As technology improves and AUV dives become longer, it becomes increasingly inefficient to require a subsequent dive to follow up on interesting features. Our approach to CoExploration specifically addresses the nested survey methodology.

Enhanced autonomy would also serve to reduce the need for multiple dives, and work in this domain has and will continue to yield incremental advances, but there are important practical reasons why progress has not been faster. The deep ocean is an expensive environment within which to test new algorithms for autonomous exploration. It is complex, unstructured, and largely unknown, so simulation environments do not yet capture sufficient detail for results to transfer directly into the real world. CoExploration, as a general approach, has the potential to not only improve the efficiency of deep-sea exploration as it exists now, but also to accelerate the deployment of more advanced autonomy. The CoExploration paradigm will allow for the incremental deployment of subsea autonomy by initially relying on humans for, e.g., the labeling of clustered data, indicating interest, or approving actions. The extent to which this is possible depends on the capacity of the acoustic link and the ability to algorithmically summarize large volumes of data into information that can be transmitted acoustically. Naturally a positive feedback exists—enhanced subsea autonomy will increasingly be able to predict data of utility to operators and likewise to abstract low level data into more compact information.

The CoExploration approach relies on bidirectional communication between a robotic explorer and its operators. In the ocean, that pathway is inherently limited by the physics of acoustic propagation. A large body of literature exists concerning the design of acoustic modems and signal processing to maximize throughput and, likewise, concerning the compression of data to more efficiently utilize available throughput. AUVs have long used acoustic modems, e.g., [Freitag et al., 2005] to transfer selected portions of data to the surface for human operators to review. The most common data are navigation and control system information—data that allow operators to monitor system performance and potentially to correct problems or modify missions on-the-fly. Engineering data and commands are typically telemetered in compact stand-alone packets (e.g. [Stokey et al., 2005], [Schneider et al., 2015]). This approach emphasizes reliability but applies poorly to payload data products (maps, time series measurements) whose value is limited if messages received topside consist only of the most recent raw measurement.

Progressive encodings provide a means for operators to request finer detail or enhanced resolution without wasting previously transmitted data. Work by [Murphy et al., 2013] demonstrates how progressive encodings can yield more effective use of an acoustic link in the context of scientific or payload data including images and time series data. Our work extends the same essential idea to bathymetry, and likewise applies to time series data and imagery. The image transfer pipeline described in [Murphy et al., 2013] is directly antecedent to the pipeline employed here.

Image compression, often leveraging widely used compressed formats (e.g., wavelet-based JPEG 2000 compression), has received considerable attention in the underwater context. In practice, the time required to transmit an image over the km-scale distances relevant to deep-sea exploration, with sufficient resolution to yield discernible features, might be several minutes—[Murphy et al., 2013] report 15 minutes for a progressively encoded 1024 x 1024 color image in real world conditions with significant packet loss. Multibeam and sidescan sonar can be processed subsea into images and handled similarly, although multibeam bathymetry in particular is amenable to other representations, e.g., multi-resolution grids [Maleika and Forczmański, 2019]. The use of quadtree and octree grids for representing seafloor bathymetry goes back at least to [Fairfield et al., 2007]. Herein we leverage the quadtree representation as a progressive encoding. Time series data, e.g., from chemical sensors, is typically much lower in volume, but often the data of interest are limited to short segments of outliers. Techniques suited to the preservation of these rare, high-frequency features were developed using a wavelet compression approach by [Murphy et al., 2013].

To our knowledge, the first use of the term CoExploration in a marine robotics context appears in [Girdhar et al., 2019] who describe an algorithm that autonomously generates a semantic map of the environment from imagery collected on board, transmits this highly compressed information to operators, and relies on operators to communicate scientific interest in the context of the map to guide subsequent path planning. This state of the art system is focused on seafloor imagery and has been deployed in a coral reef environment. Our focus here is on human-guided transmission of relatively unprocessed data, a comparatively primitive approach, but the advantage is immediate applicability to existing paradigms for deep-sea exploration. We anticipate that both the communications layer described herein and the practical lessons learned will facilitate the deployment of more advanced approaches.

III. Methods

The goal of the CoExploration toolbox is to use any acoustic bandwidth not required for operational data to transmit scientifically-actionable data. While each data modality is an independent tool, they were designed to operate similarly and provide similar interfaces to the underlying acoustic communications utility.

For each data modality, there is a subsea "sender" process that handles incoming data from the sensors and chooses what to transmit acoustically based on a default policy that can be overridden by input from topside scientists. The default will attempt to transmit a synoptic view of the data, prioritizing coverage over resolution. The scientist can send requests prioritizing specific images or regions at a desired resolution.

The data is transmitted acoustically using a separate communications manager that handles packing ROS messages, prioritizing different types of data (e.g. guaranteeing that vehicle status messages have priority over science data) and managing the TDMA schedule.

Topside, a "receiver" process aggregates the incoming data and makes it available to the GUI. It also handles turning human input from the GUI into requests for additional data. Finally, a lightweight GUI displays the received data and allows the human to request a prioritization (spatially, temporally, or by image ID).

These tools use the Robot Operating System (ROS) [Quigley et al., 2009] as a robotics middleware. ROS (along with other message-passing middleware solutions, like LCM [Huang et al., 2010] and MOOS [Newman, 2008]) encourages breaking a complicated system down into atomic chunks of functionality with defined interfaces, which makes it easier to share utilities between different robots. In the best case, additional functionality can be incorporated via simple modifications to configuration files (launch files in ROS). Implementing the CoExploration toolbox as a set of ROS nodes is key to its potential adoption in other systems, as ROS enables interoperability and portability. ROS is the de facto middleware of choice for academic robotics research and provides a more full-featured set of visualization, simulation, and debugging tools than either LCM or MOOS. ROS is increasing in popularity in the maritime robotics community, and is in operational use on vehicles including: Remus AUVs [Gallimore et al., 2018], Sentry [Vaughn et al., 2018], and Autosub [Phillips et al., 2020]. Additionally, there is active development of ROS-compatible simulation utilities for marine vehicles, including Project Dave [Zhang et al., 2022], simulation of MBARI's LRAUV vehicles in Gazebo [Open Robotics, 2022], and features being added into the core Gazebo simulation engine in support of marine vehicles (hydrodynamics, added mass, buoyancy). Bridge utilities exist to enable interoperability with other middleware systems such as MOOS, and these can be used to allow CoExploration to run on robots based on those systems.

A. Multibeam map transfer

The operational concept of nested surveys requires the ability to transfer a representation of the bathymetry data collected by a sea-floor mapping AUV during a dive, in order for topside scientists to target follow-up camera surveys later in the same dive. This required developing both the capability to generate a map in real time from sonar data and a scheme for acoustically transmitting the resulting map.

The octomap library and associated ROS wrappers allow out-of-the-box 3D mapping, using a probabilistic voxel grid to explicitly represent both obstacles and known free space[Hornung et al., 2013]. We considered using it, but there were a number of ways that it wasn't particularly suited to our use case: it is limited to a fixed region around the origin; and the voxels are cubes – there is no way to obtain 1 cm vertical resolution on a map with a horizontal resolution of 1 m.

Instead, we chose to implement our own mapping library, based on a quadtree. In a quadtree, each cell contains data fields as well as pointers to its 4 children. They are a natural data structure for storing a 2.5D grid where the extent is not known a priori, or where the shape of the surveyed area does not approximate an axis-aligned grid. Most importantly for this project, the quadtree naturally represents the topography in a hierarchical data structure that makes it simple to query lower-resolution maps, and (with the addition of appropriate metadata in each map cell) it is possible determine which subtrees have been updated and need to be re-transmitted.

In addition to pointers to its children, each node in our quadtree implementation stores the most recent timestamp at which it (or any of its children) were updated with new data, and the most recent timestamp that its data has been transmitted. Elevation is represented as the mean elevation of all sonar returns within that cell, along with a count of how many points have contributed to the mean. Growing the quadtree to extend its spatial coverage simply requires adding

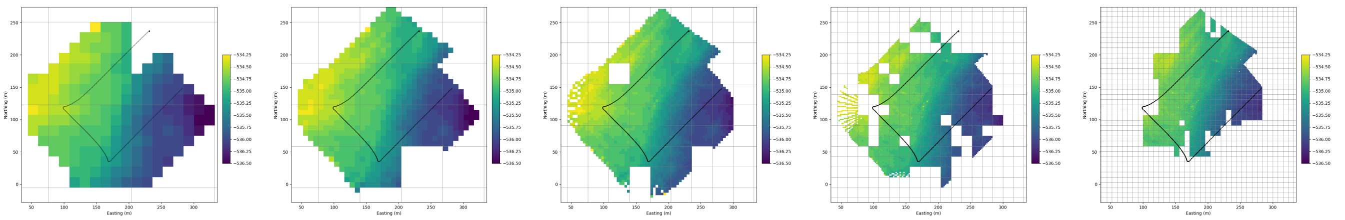

Fig. 2. Multiple resolutions of the quadtree are transmitted acoustically and received topside, starting with 16 m grid cells (far left), working down through 8 m, 4 m, 2 m and finally 1 m grid cells (far right). In each image, the light grey grid shows what region fits into an atomic update (80 or 144 bytes). The grids in this figure show data from an actual dive: data that is streamed as "background data" is not acked, and if the acoustic packet doesn't decode, it will result in a gap in the map unless the operator requests re-transmission. This map does not reflect real-time streaming capability – as this was a region of interest, the operators requested additional resolution after the survey was completed and the robot had moved on to other tasks.

a new root node to the tree, one of whose children is the old root.

Updates to the quadtree are transmitted as *subgrids* – a set of nodes at a given level of the tree with a common root N levels up. For this work, we used 8x8 subgrids, which can be represented using 80 or 144 bytes, depending on the terrain. A single bit in the subgrid encoding indicates whether one or two bytes are required to encode elevations as a delta above the subgrid's minimum.

The default policy is to attempt to transmit all data once, starting with the lowest resolution layer. Within each layer, subgrids are selected for transmission if they haven't been sent since they were last updated and their last update was at least 20 seconds ago (to prevent the same subgrid being queued repeatedly while the robot is surveying that region). No attempt is made to automatically fill in gaps in the topside map resulting from the lossy nature of acoustic communications – updates to the quadtree are self-contained, and the human operator can ask for a region to be filled in if desired. Figure 2 shows 5 levels of the same map, as received topside during an actual dive. Blank squares are the result of dropped packets.

The topside operator requests data by selecting a region of interest in the GUI along with the desired resolution. Instead of explicitly acking each update, a request for additional data contains a list of subgrids along with the most recent timestamp (if any) for which their data has been received topside. The subsea process will only re-send data in the requested regions if there was no topside timestamp or if the subsea map has been updated more recently.

This mapping pipeline is agnostic as to the specific multibeam used – it supports the ROS `sensor_msgs/PointCloud2` data type, and has been used with data from multiple instruments: Sentry's legacy Reson multibeam (converting the s7k log files to a rosbag); Sentry's new Kongsberg EM2040 multibeam[1]; data collected by a Norbit multibeam mounted on the Pontoon of Science[2] [Krasnosky et al., 2021]; and a Norbit multibeam on NUI[3].

[1] Kongsberg ROS driver: https://bitbucket.org/whoidsl/ds_kongsberg

[2] `ds_ros` Norbit driver: https://bitbucket.org/whoidsl/ds_norbit_mb

[3] Norbit ROS driver: https://bitbucket.org/croman_and_the_barbarians/norbit

B. *Progressive image transfer*

The desire to transmit images mid-dive motivates the solution presented here where imagery is sent in packets which are progressively reassembled to create a viewable "partial" image (either lower quality or lower resolution). Further packets can be sent and the image re-rendered at higher resolution or quality without re-transmission of existing information.

While the basic concept of progressive imagery transmission on marine vehicles has been demonstrated before (see [Murphy et al., 2013]), this work builds on those ideas to be more readily accessible to numerous marine robotics applications. This system is more accessible than past efforts for reasons that are both technical (C++ core library designed for reuse and documented) and non-technical (code released under the permissive BSD license and gall dependencies are open source distributions without onerous copyright or patent restrictions).

Images generated by the sensor are stored in the widely used TIFF image format. The TIFFs are then read by the *progressive_imagery* library and compressed using the JPEG2000 format (which supports a configurable progression ordering, e.g. layer/quality or resolution) by the OpenJPEG library [Image and Signal Processing Group, Université de Louvain,]. At this point the JPEG2000 codestream is packetized into fixed frames suitable for marine links, and the metadata efficiently compressed using the Dynamic Compact Control Language (DCCL) [Schneider et al., 2015], which provides substantial reduction of the codestream metadata. Finally, the packets are queued and transmitted as the comms link is available.

On the receiver side, each image packet has its metadata decompressed and is reassembled into a (partial) JPEG2000 codestream, which is then decompressed and written to disk as a TIFF image for use by the operator. This provides the ability for the operator to make immediate decisions regarding the AUV mission or data prioritization. Figure 3 provides an example of how image resolution improves as a function of bytes transferred. In order to allow for operations with a lossy communications channel, received packets are acknowledged, and the sender will re-transmit packets for which it has not received an ack.

The key design choices that make the progressive imagery tool suitable for marine operations include:

- Maximum packet size is predictable and configurable for

use with a particular maximum transmission unit (MTU), such as those found on acoustic modems. The MTUs of acoustic links vary; the WHOI Micromodem offers MTUs from 32 bytes to 2 kbytes, depending on the chosen data rate.
- Metadata is highly compressed using DCCL to reduce packet overhead.
- The library provides a simple API that can be wrapped in the robotic middleware of choice (thus far ROS and Goby3 have been used supported).
- A queuing protocol allows the system to automatically send each image to a configured quality (SNR) or resolution set-point and then move on to the next image. If, based on this small amount of image data, the operator wants to see more from a particular image, they can send an updated set of "queue goals" which will request higher resolution or quality to be sent for one or more images, up to and including the full image data.

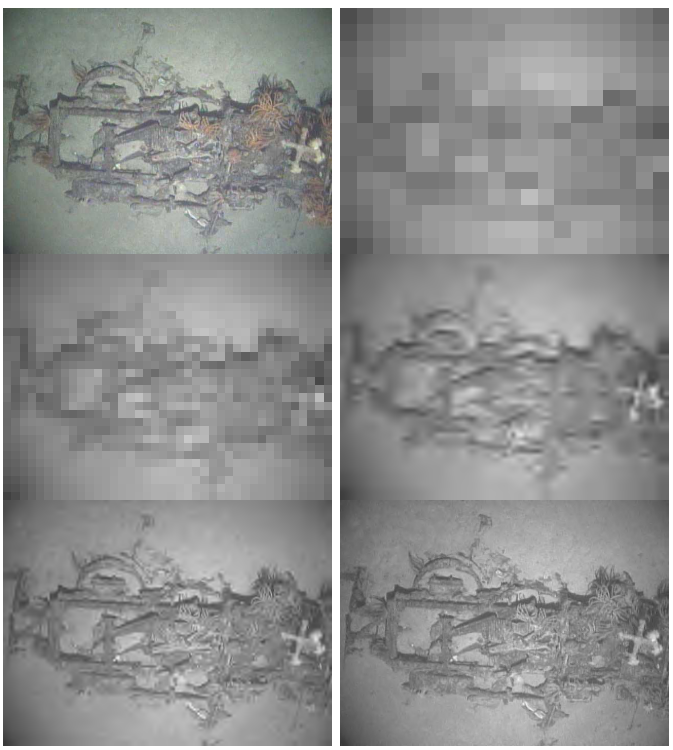

Fig. 3. Reconstructed image quality as a function of data transferred. **(top left)** Original 480 x 360 color image. **(top right)** After 1 frame / 214 bytes. **(middle left)** After 2 frames / 382 bytes **(middle right)** After 3 frames / 626 bytes. **(bottom left)** After 16 frames / 3632 bytes, or two rate-five Micromodem packets (requiring $\sim$ 40 seconds of transmission time at 10 kHz with 2 kHz bandwidth). **(bottom right)** Final image, which would involve transferring 20,997 bytes over 93 frames / 12 packets in an idealized loss-free scenario.

C. Acoustic Channel Management

The tools described in Sections III-A and III-B assume that there is a way to acoustically transmit ROS messages between the vehicle and the topside scientists. A typical acoustic communications stack will include functionality for:

1) Managing the TDMA cycle for when each modem transmits
2) Efficiently encoding ROS messages
3) Queuing and prioritizing messages for transmission, typically on a per-topic basis
4) Decoding and republishing ROS messages received in the modem data payload
5) A driver for the modem

CoExploration was designed to be agnostic to the acoustic communications stack used, and has been tested with two different systems: a standalone communications manager, designed as part of this work; and `ros_acomms` [Gallimore et al., 2022], a system designed by the Acoustic Communications Group at WHOI.

For the work presented in this paper, we used `ros_acomms` as the communications stack. A complete description of `ros_acomms` is given in [Gallimore et al., 2022], so here we only summarize the key features used for CoExploration. With `ros_acomms`, any ROS message can be sent from one system to another over the acoustic link. Messages are efficiently encoded and marshalled, and multiple messages are automatically packed into single acoustic packets when possible.

`ros_acomms` provides a message queuing subsystem that assigns priorities to messages. These priorities can be changed both at configuration-time via a YAML file and at run-time via a topside GUI developed by the CoExploration project. This prioritization capability is used by the topside scientists to select which categories of data should be sent from the vehicle at different stages of the dive. Since all messages to and from the vehicle use the same queue system, critical vehicle telemetry and commands are prioritized, and any remaining space in the acoustic packets is filled with scientific data.

To support the CoExploration application, a shim node was built to interface between the `ros_acomms` message-based architecture and the service-call-based background data transfer method used by the progressive imagery and quadtree utilities. It is responsible for monitoring the queue lengths and requesting enough background data to fill the next transmission. Given the interactive nature of these utilities, we only want to generate as much data as can be consumed by the acoustic stack in a given transmission window – any additional data will either be dropped or queued up, leading to potentially significant lag in responding to the operator's requested priorities.

Furthermore, the CoExploration and `ros_acomms` system is transparent to the modem hardware used. Integrating a different modem into the system simply requires a software driver node that implements standardized ROS interfaces for sending and receiving packets. In addition to multiple dives using the WHOI Micromodem2, we used the Sonardyne SMS protocol on one dive to demonstrate cooperative operation with other vehicles that only had Sonardyne modems.

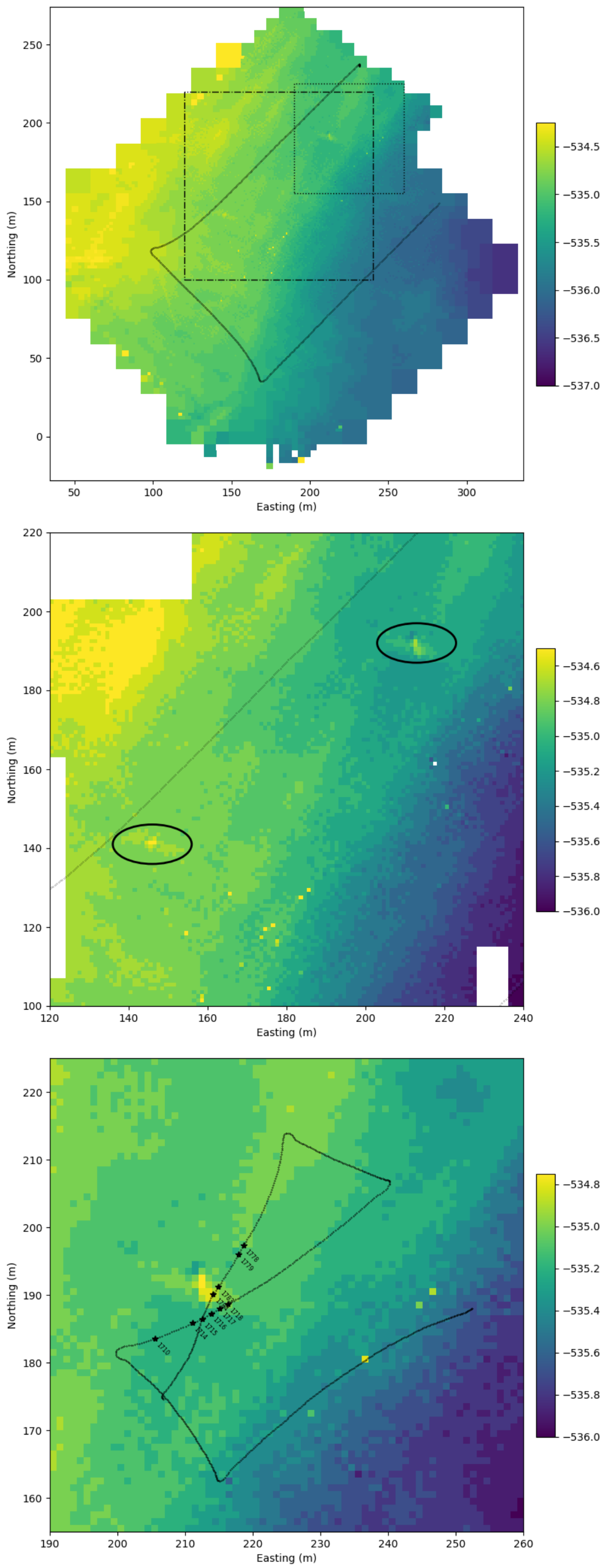

Fig. 4. Bathymetry received over the course of NUI dive 040. **(top)** Final multibeam map at 1 m resolution, showing NUI's path for the two survey lines. (See Figure 2 for the individual layers at different resolutions.) The dash-dotted (dotted) rectangle indicates the region shown in the middle (bottom) subfigure. **(middle)** Zoom in on multibeam map showing suspected locations of the two vehicles, which are circled. **(bottom)** Camera survey across vehicle 32. Stars indicate the locations of camera images that were queued for transfer.

IV. Results

The CoExploration toolbox was demonstrated on the NUI vehicle [Jakuba et al., 2018] during 5 separate dives on a 2022 cruise aboard R/V Nautilus, where it routinely telemetered up bathymetry, time series data, and images. NUI can be optionally tethered by a kms-long hair-thin fiber optic microtether that enables high bandwidth real time access to vehicle data without impairing vehicle mobility. This capability sped development by enabling us to tune, reconfigure, and debug on-the-fly rather than strictly between dives. All data presented subsequently was telemetered acoustically via the CoExploration pipeline.

In this section, we focus on NUI dive 040, where progressive transfer of bathymetry enabled an adaptive camera survey. For this dive, we knew that there were multiple "wrecks" (submerged cars) in a region. The dive started with a multibeam survey at 30 m altitude, with the goal of identifying the cars as elevation anomalies against a flat seafloor. Due to operational constraints, the R/V Nautilus was holding station ∼1.5 water depths away from the survey site, which contributed to poor acoustic performance during the survey. After the sonar survey, human pilots (controlling NUI via the microtether) explored the area in an attempt to locate both cars based on approximate GPS coordinates, but only found the southwestern one.

The bathymetric map transferred in real-time was insufficient to identify locations of interest; instead, we carried out a data mule-ing step where NUI drove closer to the ship where the near-vertical acoustic channel allowed more reliable communications. While data mule-ing added time to the mission, it was still dramatically more efficient than recovering the vehicle, performing a maintenance cycle, and re-launching NUI. Figure 2 shows each layer of the resulting map as it was received topside, and Figure 4 (top) shows the resulting compiled map. These features were small enough that they were not discernible at any resolution coarser than 2 m cell sizes, and only at 1 m resolution was it easy to distinguish them from the single-cell outliers that appeared due to noisy returns at the edge of the sonar's beam pattern.

After the sonar map was acoustically transmitted topside, operators commanded a survey at camera altitude crossing the northeastern anomaly (Figure 4 (bottom)). Based on NUI's dead-reckoned navigation solution and topside USBL tracking, operators requested an image (#1718) based on the time at which we expected to have flown over the car. Tracking was not perfect, and the image did not contain the full car, but there was a hint of something in the corner of the frame, allowing us to step back in time until we found the car. The Progressive Imagery pipeline only required one or two TDMA cycles with successfully decoded packets per image for the topside operators to determine whether it contained the car. Figure 5 shows all of the images that were received topside over the course of the dive, and their approximate locations are shown in Figure 4 (bottom). One requested image was not received, due to interactions between a very lossy acoustic channel and the way the progressive imagery

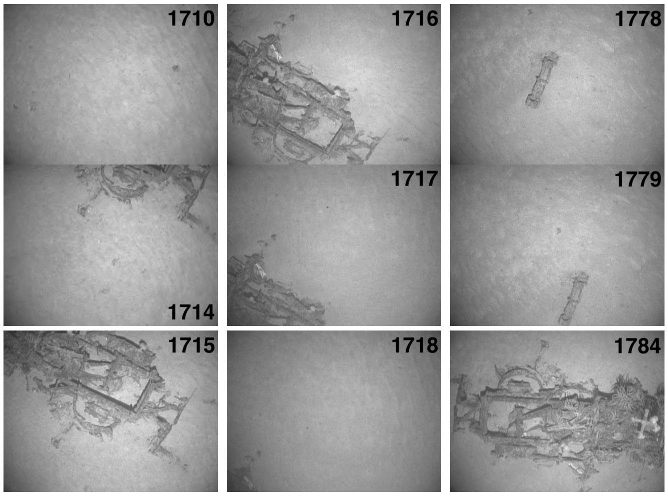

Fig. 5. Images transferred acoustically during the dive; Figure 4 shows the location of each image along the camera survey.

pipeline and `ros_acomms` handle acks. We suspect that the configured message queue depth was exceeded, which left the image unable to be reassembled. This suggests a bug in the implementation.

V. Discussion / future work

`ros_acomms` was originally designed to support a network of fully autonomous AUVs, rather than human-in-the-loop communication. As such, it is configured at the start of a mission, after which it is not possible to update the TDMA schedule. However, in the CoExploration use case, this is a highly desirable feature. NUI hosts multiple frequencies of acoustic modems, optimized for communication at different ranges. Each modem had a separate instance of the `ros_acomms` stack controlling its TDMA schedule and message queues. Rather than establishing a TDMA schedule that leaves space for all of them (and thus requires a several-minute period), we would have preferred to enable/disable them over the course of the dive to continuously optimize throughput.

Relatedly, current operation of the CoExploration system rewards an expert operator who is tracking what has been requested, what has been received, and the quality of the acoustic connection. Adoption by more operationally-focused AUVs will require a more polished interface that presents this data to the user. Additionally, we need a unified display of all acoustically-transmitted data, providing spatial and temporal awareness, in context with other data from the vehicle, topside tracking, and scientific targets.

Typical AUV survey operations aren't particularly dependent on precise positioning during a dive – instead, it is more important that a survey be flown evenly spaced with smooth tracklines, and that the vehicle's trajectory can be reconstructed in post-processing after the dive, with the aid of topside USBL measurements. For adaptive nested surveys, this was not sufficient – we needed to be able to go back to a precise spot on an earlier survey, correcting for navigation drift in the vehicle's dead-reckoned estimate. For the work in this paper, this correction was done in an ad-hoc fashion, by expert human operators. Operational use of real-time bathymetric maps will require a more polished solution to the state estimation problem, since it requires both smooth trajectories without jumps (in order for overlapping regions of the map to line up) and the ability to determine globally-precise coordinates.

VI. Code Availability

All software developed as part of the CoExploration project has been publicly released under the BSD license. Progressive Imagery is a standalone project (https://bitbucket.org/whoidsl/progressive_imagery), while the rest of the utilities are in a single repository (https://bitbucket.org/whoidsl/coexploration). Additionally, we have prepared a repository with a README, the data presented in this paper, and appropriate launch files to run progressive image transfer and multi-resolution bathymetric transfer in simulation with the data presented in this paper: https://github.com/lauralindzey/nui_coex_auv_2022.

Acknowledgments

The authors thank the Ocean Exploration Trust and the captain and crew of the R/V Nautilus who provided excellent support for NUI operations. Additionally, we thank the NUI field team: Molly Curran, Casey Machado, Robin Littlefield, Victor Naklicki, and Allisa Dalpe. Eric Gallimore, Dennis Giaya, and Caileigh Fitzgerald from the WHOI Acoustic Communications Group greatly assisted with our adoption of `ros_acomms`. We are grateful for the opportunity to trial CoExploration at sea with the AUV Sentry prior to the work reported here and wish to specifically acknowledge the support of Stefano Suman and Sean Kelley in that endeavor. We further acknowledge ongoing work by Timothy Joyce to integrate CoExploration utilities into Sentry's topside visualization software. LL thanks Ian Vaughn for his impassioned discussions about bathymetric multibeam, and Zac Berkowitz for his assistance getting started with acoustic communications. We gratefully acknowledge funding from the NOAA office of Ocean Exploration and Research under grant NA17OAR0110207 and from the NOAA Ocean Exploration Cooperative Institute (NA19OAR4320072).

References

[Brothers et al., 2013] Brothers, L., Van Dover, C., German, C. R., Kaiser, C. L., Yoerger, D., Ruppel, C., Lobecker, E., Skarke, A., and Wagner, J. (2013). Evidence for extensive methane venting on the southeastern us atlantic margin. *Geology*, 41(7):807–810.

[Fairfield et al., 2007] Fairfield, N., Kantor, G., and Wettergreen, D. (2007). Real-time slam with octree evidence grids for exploration in underwater tunnels. *Journal of Field Robotics*, 24(1-2):03–21.

[Fisher et al., 2014] Fisher, C. R., Hsing, P.-Y., Kaiser, C. L., Yoerger, D. R., Roberts, H. H., Shedd, W. W., Cordes, E. E., Shank, T. M., Berlet, S. P., Saunders, M. G., et al. (2014). Footprint of deepwater horizon blowout impact to deep-water coral communities. *Proceedings of the National academy of sciences*, 111(32):11744–11749.

[Freitag et al., 2005] Freitag, L., Grund, M., Singh, S., Partan, J., Koski, P., and Ball, K. (2005). The whoi micro-modem: An acoustic communications and navigation system for multiple platforms. In *Proceedings of OCEANS 2005 MTS/IEEE*, pages 1086–1092. IEEE.

[Gallimore et al., 2022] Gallimore, E., , Giaya, D., Miller-Klugman, B., Griffen, K., Fitzgerald, C., and Freitag, L. (2022). Ros message transport over underwater acoustic links with ros_acomms. In *2022 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*. IEEE. in review.

[Gallimore et al., 2018] Gallimore, E., Stokey, R., and Terrill, E. (2018). Robot operating system (ros) on the remus auv using recon. In *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pages 1–6. IEEE.

[German et al., 2008] German, C. R., Yoerger, D. R., Jakuba, M., Shank, T. M., Langmuir, C. H., and Nakamura, K.-i. (2008). Hydrothermal exploration with the autonomous benthic explorer. *Deep Sea Research Part I: Oceanographic Research Papers*, 55(2):203–219.

[Girdhar et al., 2019] Girdhar, Y., Cai, L., Jamieson, S., McGuire, N., Flaspohler, G., Suman, S., and Claus, B. (2019). Streaming scene maps for co-robotic exploration in bandwidth limited environments. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7940–7946. IEEE.

[Hornung et al., 2013] Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., and Burgard, W. (2013). Octomap: An efficient probabilistic 3d mapping framework based on octrees. *Autonomous robots*, 34(3):189–206.

[Huang et al., 2010] Huang, A. S., Olson, E., and Moore, D. C. (2010). Lcm: Lightweight communications and marshalling. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4057–4062. IEEE.

[Image and Signal Processing Group, Université de Louvain,] Image and Signal Processing Group, Université de Louvain. OpenJPEG: An open-source JPEG 2000 codec written in C.

[Jakuba et al., 2018] Jakuba, M. V., German, C. R., Bowen, A. D., Whitcomb, L. L., Hand, K., Branch, A., Chien, S., and McFarland, C. (2018). Teleoperation and robotics under ice: Implications for planetary exploration. In *2018 IEEE Aerospace Conference*, pages 1–14. IEEE.

[Kaiser et al., 2016] Kaiser, C., Yoerger, D., Kinsey, J., Kelley, S., Billings, A., Fujii, J., Suman, S., Jakuba, M., Berkowitz, Z., German, C., and Bowen, A. (2016). The design and 200 day per year operation of the autonomous underwater vehicle Sentry. In *Autonomous Underwater Vehicles 2016, AUV 2016*.

[Krasnosky et al., 2021] Krasnosky, K., Roman, C., and Casagrande, D. (2021). A bathymetric mapping and slam dataset with high-precision ground truth for marine robotics. *The International Journal of Robotics Research*, page 02783649211044749.

[Maleika and Forczmański, 2019] Maleika, W. and Forczmański, P. (2019). Adaptive modeling and compression of bathymetric data with variable density. *IEEE Journal of Oceanic Engineering*, 45(4):1353–1369.

[Murphy et al., 2013] Murphy, C., Walls, J. M., Schneider, T., Eustice, R. M., Stojanovic, M., and Singh, H. (2013). Capture: A communications architecture for progressive transmission via underwater relays with eavesdropping. *IEEE Journal of Oceanic Engineering*, 39(1):120–130.

[Newman, 2008] Newman, P. M. (2008). Moos-mission orientated operating suite. Technical report, Department of Engineering Science, University of Oxford.

[Open Robotics, 2022] Open Robotics (2021–2022). LRAUV Simulation. https://github.com/osrf/lrauv/.

[Phillips et al., 2020] Phillips, A. B., Kingsland, M., Linton, N., Baker, W., Bowring, L., Soper, S., Roper, D. T., Johnson, A., Morrison, R., Ciaramella, K., et al. (2020). Autosub 2000 under ice: Design of a new work class auv for under ice exploration. In *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, pages 1–8. IEEE.

[Purcell et al., 2011] Purcell, M., Gallo, D., Packard, G., Dennett, M., Rothenbeck, M., Sherrell, A., and Pascaud, S. (2011). Use of remus 6000 auvs in the search for the air france flight 447. In *OCEANS'11 MTS/IEEE KONA*, pages 1–7.

[Quigley et al., 2009] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A. Y., et al. (2009). Ros: an open-source robot operating system. In *ICRA workshop on open source software*.

[Schneider et al., 2015] Schneider, T., Petillo, S., Schmidt, H., and Murphy, C. (2015). The dynamic compact control language version 3. In *OCEANS 2015-Genova*, pages 1–7. IEEE.

[Stokey et al., 2005] Stokey, R. P., Freitag, L. E., and Grund, M. D. (2005). A compact control language for auv acoustic communication. In *Europe Oceans 2005*, volume 2, pages 1133–1137. IEEE.

[Vaughn et al., 2018] Vaughn, I., Suman, S., Berkowitz, Z., Vaccaro, J., Kelley, S., Fujii, J., Jakuba, M., Howland, J., Whitcomb, L., and Kaiser, C. (2018). Transitioning to open source at 6000m. In *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pages 1–6. IEEE.

[Yoerger et al., 2007] Yoerger, D. R., Jakuba, M., Bradley, A. M., and Bingham, B. (2007). Techniques for deep sea near bottom survey using an autonomous underwater vehicle. *The International Journal of Robotics Research*, 26:41–54.

[Zhang et al., 2022] Zhang, M. M., Choi, W.-S., Herman, J., Davis, D., Vogt, C., McCarrin, M., Vijay, Y., Dutia, D., Lew, W., Peters, S., and Bingham, B. (2022). Dave aquatic virtual environment: Toward a general underwater robotics simulator. In *2022 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*. IEEE. in review.