SCIENCE AND TECHNOLOGY ORGANIZATION
CENTRE FOR MARITIME RESEARCH AND EXPERIMENTATION

Reprint Series **CMRE-PR-2019-051**

Continuous active sonars for littoral undersea surveillance

Andrea Munafò, Gaetano Canepa, Kevin D. LePage

May 2019

Originally published in:

IEEE Journal of Oceanic Engineering (Early Access), July 2018, pp. 1-15,
doi: 10.1109/JOE.2018.2850578

About CMRE

The Centre for Maritime Research and Experimentation (CMRE) is a world-class NATO scientific research and experimentation facility located in La Spezia, Italy.

The CMRE was established by the North Atlantic Council on 1 July 2012 as part of the NATO Science & Technology Organization. The CMRE and its predecessors have served NATO for over 50 years as the SACLANT Anti-Submarine Warfare Centre, SACLANT Undersea Research Centre, NATO Undersea Research Centre (NURC) and now as part of the Science & Technology Organization.

CMRE conducts state-of-the-art scientific research and experimentation ranging from concept development to prototype demonstration in an operational environment and has produced leaders in ocean science, modelling and simulation, acoustics and other disciplines, as well as producing critical results and understanding that have been built into the operational concepts of NATO and the nations.

CMRE conducts hands-on scientific and engineering research for the direct benefit of its NATO Customers. It operates two research vessels that enable science and technology solutions to be explored and exploited at sea. The largest of these vessels, the NRV Alliance, is a global class vessel that is acoustically extremely quiet.

CMRE is a leading example of enabling nations to work more effectively and efficiently together by prioritizing national needs, focusing on research and technology challenges, both in and out of the maritime environment, through the collective Power of its world-class scientists, engineers, and specialized laboratories in collaboration with the many partners in and out of the scientific domain.

NOTE: The CMRE Reprint series reprints papers and articles published by CMRE authors in the open literature as an effort to widely disseminate CMRE products. Users are encouraged to cite the original article where possible.

Continuous Active Sonars for Littoral Undersea Surveillance

Andrea Munafò, *Member, IEEE*, Gaetano Canepa, and Kevin D. LePage

***Abstract*—Recent advances in transducer and computing technology have pushed the concept of continuous active sonar (CAS) or high duty cycle sonar as an area of interest for application to antisubmarine warfare. Unlike conventional pulsed active sonars, CAS processing aims at detecting echoes while transmitting with a nearly 100% duty cycle. This paper describes the signal processing chain developed at the Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy, for real-time CAS processing within an autonomous networked multistatic sonar system. The algorithm uses subband processing, which can potentially provide a higher target update rate than the traditional pulsed active sonar, while maintaining the same search radius. The higher rate of contact information can improve target tracking performance. Performance results are given from the COLLAB–NGAS14 sea trial where the CAS processor was deployed on two autonomous underwater vehicles (AUVs) acting as receivers of the CMRE experimental multistatic demonstrator. Results show the feasibility of the CAS concept in littoral scenarios, using AUVs as real-time receivers.**

***Index Terms*—Antisubmarine warfare (ASW), autonomous underwater vehicles (AUVs), continuous active sonars (CAS), multistatic sonars, signal processing.**

I. Introduction

RECENT advances in sonar hardware, with the availability of larger sensor dynamic ranges, and the increase in compute power, which allows to process larger bandwidth signals in real time, have paved the way to in-the-field implementation of continuous active sonar (CAS) or high duty cycle (HDC) sonar for antisubmarine warfare (ASW) in shallow waters. Typically, this means repeating the transmission of a long-duration waveform, without pause, with a repetition rate similar to that used with the traditional pulsed active sonar (PAS). In PAS systems, a short monochromatic continuous wave or a frequency modulated (FM) pulse is often transmitted, followed by a relatively long period of time during which no transmission occurs and the receiver listens for echoes. In contrast, CAS systems transmit and receive echoes simultaneously, and the resulting continuous target ensonification can potentially improve the ASW detection and tracking performance.

A schematic of the two approaches is shown in Fig. 1. A PAS-based system transmits a short burst, followed by a long listening time. The short pulse has been traditionally used because older sonar systems with a limited analog-to-digital (AToD) receiving dynamic range would become saturated during the transmission, while the duration of the transmission-free listening time determines the maximum distance at which the system can unambiguously associate object detections to range with the ping energy which produced them.

The typical length for the transmitted burst is around 1 s, while the reception time is in the order of tens of seconds. As a result, in a PAS sonar, a potential target is ensonified only during a short period of time, and it might provide a new detection opportunity only after the repetition cycle is completed. In contrast, a CAS system with a sufficiently high dynamic range in the receiver is able to transmit and receive continuously, sweeping across several hundred hertz in tens of seconds, and after each transmission cycle, it immediately begins the next one.

This leads to the two main ways in which a CAS system might improve the ASW sonar performance. First, target detection probability depends on the transmitted energy. Extending the pulse duration while maintaining a constant source level increases the total energy (pulse energy is maximized with a 100% duty cycle), and hence, in noise-limited environments, enlarges the detection probability.

Second, rather than processing the entire pulse coherently, linear FM (LFM) sweeps or alternative suitable waveforms can be segmented and treated as a series of short, noninterfering pulses, which are processed individually. This type of subbanding makes it possible to avoid the processing of signals characterized by high time–bandwidth products, which might not be able to obtain the theoretically predicted processing gain due to Doppler mismatch and, especially in shallow waters, due to short coherence time [1]. This offers the important advantage of increasing the update rate of sonar contacts while maintaining the same pulse repetition interval (PRI) and corresponding search radius.

Some additional advantages of CAS have been discussed in [2]. The author noted that instead of keeping the source level constant and increasing transmission energy, the CAS transmission could have a 10–20-dB reduction in source level and maintain the same transmission energy as PAS systems. This

Manuscript received April 28, 2017; revised February 1, 2018 and April 24, 2018; accepted June 14, 2018. This work was supported by the NATO Allied Command Transformation, Collaborative Antisubmarine Warfare Programme under Project "System concepts for littoral surveillance" (Project 1). This work was presented in part at the 2015 OCEANS Conference, Genova, Switzerland, May 18-21, 2015. *(Corresponding author: Andrea Munafò.)*

Associate Editor: L. Culver.

A. Munafò was with the NATO STO Centre for Maritime Research and Experimentation, La Spezia 19126, Italy. He is now with the Marine Autonomous and Robotic Systems, National Oceanography Centre, Southampton SO14 3ZH, U.K. (e-mail: andrea.munafo@noc.ac.uk).

G. Canepa and K. D. LePage are with the NATO STO Centre for Maritime Research and Experimentation, La Spezia 19126, Italy (e-mail: Gaetano.Canepa@cmre.nato.int; Kevin.LePage@cmre.nato.int).

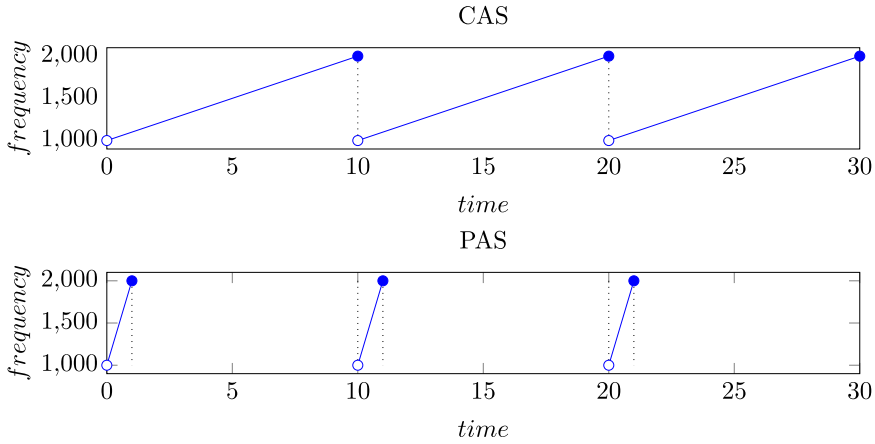

Fig. 1. Schematic spectrograms of the CAS and PAS cycles.

has several advantages: reduced power and size requirements for the amplifiers and transducers; reduced risk of transducer cavitation at shallow depth; a reduction in transmission nonlinearities; and increased transmission bandwidth available for most transducers. The gain in bandwidth is possible because most transducers do not have a spectrally flat transmission voltage response over a wide bandwidth. Therefore, by limiting the source level and the corresponding maximum voltage required, a wide bandwidth can be transmitted with a flat spectrum. The increased bandwidth is of particular interest since it is helpful for improving detection, tracking, and classification.

However, this approach, traditionally used in deep water, may be challenging in littoral areas that are under reverberation-limited conditions. CAS advantages have not yet been validated through experimentation in coastal waters where sound propagation is complex and false alarms can overwhelm active sonar systems. The 2013 Target and Reverberation Experiment (TREX13) was conducted by Canada and the United States to directly compare the CAS performance to the more mature PAS technology [2]. The experiments employed an echo repeater (ER) in a littoral environment with shallow water (less than 20 m). Recent analysis of the TREX13 data has been reported in [3] and [4].

In [5] and [6], some initial results on how the CAS processing interval used for coherent processing can be tuned to the peculiarities of the acoustic environment have been showed. Extension of existing PAS classifiers to CAS have been discussed in [7], in which the authors apply a matched filter with the full-band replica as is commonly used in PAS. This approach potentially maximizes classification performance rather than providing faster updates as in the subband approach, but may suffer from coherence loss caused by the long duration of CAS waveforms. Results on how the rapid measurement update rate of CAS can be exploited to limit the target's area of uncertainty of a tracker have been shown in [8].

It should be noted that alternative approaches to CAS processing are available. For example, matched filtering with multiple replicas adapted to specific Doppler hypotheses can be used to address Doppler losses [1], or continuous transmission frequency modulation methods [9] that heterodyne or dechirp the received signals might offer a more convenient and efficient way to accommodate a variety of update rates at the expense of processing complexity.

Regardless of the specific system used (i.e., PAS and CAS), the state of the art in an active ASW is based upon the deployment of active sonobuoys from marine patrol aircraft and marine patrol helicopters, or on the use of hull-mounted and variable depth sonar deployed from surface ships, working in concert with receivers, mounted as towed line or hull-mounted arrays on submarines and frigates [10], [11].

It has only recently been found that some pioneering works have shown that alternative approaches based on networked autonomous systems are possible [12]–[14]. The underlying idea being that many autonomous sensing units, fixed (e.g., sonobuoys) and/or mobile [autonomous underwater vehicles (AUVs)], when properly interconnected [15], [16], have the potential to achieve a network gain (i.e., maintain/improve coverage and protection) able to compensate for less capable and less expensive individual platforms. The resulting system is characterized by a greater degree of adaptation, with no single point of weakness, and where the richness of sensors enables multistatic fusion and tracking [17]. However, at the same time, when such simpler platforms are deployed, they face an exceptionally challenging scenario, whereas onboard and remote sensors collect large amount of data that have to be processed to extract the relevant information, i.e., possible targets, threats, false alarm, etc., so that local autonomous decisions can be made and (compressed) information can be shared with the rest of the network (other sensors, communication nodes, and/or the command and control). The real-time processing is the heart of AUV-based systems, bridging the sensors with the more complex adaptive behaviors. Of course, to make these decisions in real time, data processing must also be completed in real time. Vehicle size, heat generation, and power constraints are all factors that limit the computational abilities of such small assets, requiring a tradeoff between the achievable processing performance and computation time.

The work presented in this paper describes a multistatic CAS system that has been explicitly designed to be implemented on

distributed autonomous assets. More specifically, the following points have been discussed.

1) It describes the structure of a CAS system that has been designed to be deployed within the Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy, autonomous networked multistatic sonar system. The multistatic sonar is composed of an active source towed from a ship or deployed offboard and two receivers installed onboard AUVs, as well as a towed receiver array deployed from the (nearly monostatic) source ship. To work with this experimental capability, CMRE has developed, through the years, a real-time CAS processing chain, which allows immediate feedback of CAS performance during sea trails. This real-time feedback is critical to ensure that the systems are working properly and data are being collected that are suitable for thorough scientific analysis in postprocessing. The real-time capability also plays an important role in allowing CMRE to demonstrate new technology in operational settings, such as ASW exercises. This paper provides the details of CMRE's CAS processing, which is based on software developed and validated at CMRE for PAS systems [12]. Particular attention will be given to stress those parts that needed careful tailoring for implementation on embedded systems. Computational efficiency was an important consideration in the development, since an immediate goal was to run the processing in real time onboard CMRE's Ocean EXplorer (OEX) AUVs with constrained computing resources.
2) Results are given from the recent CMRE COLLAB–NGAS14 sea trial, held off the coast of La Spezia, Italy, in October 2014. The software ran in real time on the AUVs, and was able to successfully detect and track an ER towed by *NRV Alliance*, La Spezia, Italy, in a bistatic configuration. The CAS pulse, an 18-s LFM swept over the 1800–2500-Hz band, was broken into eight subbands. This provided detections on the ER at a rate six times faster than would be possible with pulsed sonars with the same PRI. From this perspective, it seems easy to see how the increased rate of information that the CAS makes available to the AUVs might, in the future, reflect on the vehicles ability to take better autonomous decisions.

This paper is laid out as follows. Section II gives an overview of the CMRE networked multistatic sonar system. Section III goes into the details of the signal processing algorithms used. Section IV describes the setup used to fit the signal processing library onboard the CMRE OEX vehicles. Section V describes the COLLAB–NGAS14 experimental setup and the results that we obtained in the field. Finally, Section VI draws some conclusions.

II. Centre for Maritime Research and Experimentation Multistatic Autonomous Network

A. Ocean Explorer Autonomous Underwater Vehicle With Slim Towed Array

The main research components of the CMRE multistatic sonar are two CMRE OEX AUVs (named OEX Harpo and OEX Groucho) used in combination with the in-house developed CMRE's SLIm Towed Array (SLITA). The OEX is a torpedo-shaped AUV of 4.5-m length and a diameter of 0.53 m (21″), which can operate at a maximum depth of 300 m. Its maximum speed when towing the array is 1.5 m·s^{-1}. The vehicles battery limit is about 16 h. The OEX is equipped with an EvoLogics acoustic modem that operates in the range 7–17 kHz, with a maximum nominal range of 8 km, and which is used for sharing data with the other vehicle and with the command and control center [18]. The OEX AUV, together with the SLITA array, is shown in Fig. 2. The SLITA array is a uniform linear array with a total of 83 hydrophones (sensitivity −201 dB ref. to 1 V/μPa), nested in four sets of 32 that can be selected at any given time, with optimal frequencies ranging from 714 to 3471 Hz. Note that since it is a linear array, it lacks the ability to resolve the left–right ambiguity [19]. The array is also equipped with three compasses and two depth sensors to help the reconstruction of the array dynamics. The total array length is 57 m, and the acoustic section starts 9 m behind the vehicle. The SLITA array described in this paper is an upgraded version of the one described in [20].

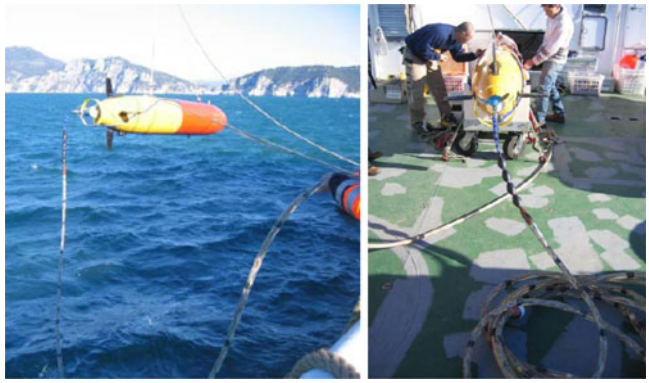

Fig. 2. OEX AUV during deployment from the *NRV Alliance* (left) and on deck (right). The SLITA is visible hanging vertically behind the AUV.

Each OEX is equipped with a main computer, which directly controls the vehicle guidance and navigation. This main computer (front-seat computer) is capable of controlling the vehicle for preplanned missions, and for receiving basic commands (e.g., abort a mission and surface) from the command and control center via acoustic communication. In accordance with the "front-seat driver/back-seat driver" paradigm [21], the OEX AUV has also a configurable payload section (back seat) that makes decisions about what commands to send to the front seat. The payload section consists of two computer systems. The first computer (slita-pc), a PC-104 board with a single core 1.4-GHz Pentium-M processor with 1 GB of RAM, is used only for data acquisition from the SLITA array. The second, more powerful, computer (backseat-pc) is used for signal processing and MOOS-IvP autonomous decision making [22], [23]. Presently, this second computer is using an AMD GX-420CA quad-core 2 GHz processor with 8 GB of RAM. Both computers run Linux operating systems making the payload section flexible for rapid development and ease of integration, although it is not a strict real-time system. In what follows, "real time"

Fig. 3. ATLAS MF source during deployment from the *NRV Alliance*.

is always considered with respect to the data being processed, i.e., processing is completed faster than new data comes in and, therefore, the processing can keep up with incoming data. The two payload computers are connected to each other via gigabit Ethernet. The LonTalk protocol is used to connect the payload to the vehicle control computer.

Using this setup, data are acquired by the slita-pc through an AToD converter board, which reads one data channel for each of the 32 hydrophones in the SLITA array. Data are archived on the local slita-pc hard drive, and transferred via the standard network file system (NFS) to the hard drive of the backseat-pc where the MOOS middleware [22], [23] and signal processing operations run.

B. ATLAS Acoustic Source

The ATLAS source, shown in Fig. 3, is a programmable towed acoustic source based on free-flooded ring technology. The source has a maximum source level of 219 dB, and it works within a programmable acoustic frequency range from 1 to 4 kHz. In our experimental setup, the ATLAS source is deployed and towed from the *NRV Alliance*, which means that the types of acoustic signals that are transmitted can be easily changed (although for this specific work, only LFM signals were considered, and other advanced waveform types will necessarily require alternate processing approaches). Accurate transmission timing and source localization are critical inputs for multistatic sonar processing. This is achieved by coupling the ATLAS source to a waveform generator that is synchronized to the GPS data provided from the ship.

III. Sonar Signal Processing

The CAS signal processing suite was built as an evolution of algorithms from CMRE's previous multistatic PAS processors [12]. The new algorithms use subband processing, as discussed in Section I. This method breaks up a long pulse into short segments that are processed individually using PAS processing. The approach of building from the existing PAS processing was, therefore, suitable, and allowed rapid development and facilitating validation of the CAS processing using the well-tested PAS software. The drawback of this approach was that the developers were limited to previous software structure, which was not always conducive to further development. The implemented signal processor is a frequency-domain conventional beamformer and a matched filter library for CAS array processing. This section gives a general description and parametrization of the algorithms employed. Details on the settings used for the experiments are given in Section V-B.

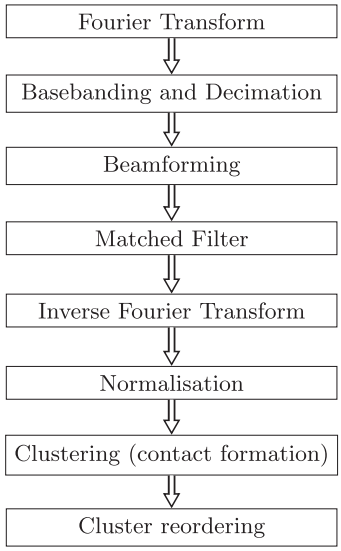

Fig. 4. CAS signal processing chain: beamforming, matched filter, normalization, and contact formation. This structure resembles more traditional PAS schemes. The contact reordering block is a peculiarity of the proposed scheme based on data file of fixed length, and it is necessary to correctly associate formed contacts to the corresponding ping. Each box is detailed in the text.

A. Data Reading, Conversion, and Detrending

Data for each hydrophone of the SLITA array are recorded into a file of length defined by the PRI at the specific sampling rate F_s by the slita-pc and made available to the processing software on the backseat-pc via the standard NFS. The signal processor detects the creation of a new file and reads it into memory. Once the file is loaded into memory, the CAS processor moves through the following stages (see Fig. 4): Fourier transform to frequency domain, basebanding and downsampling, beamforming, subband matched filtering, inverse Fourier transform to time domain, detection and cluster formation, and cluster reordering. This last stage associates all the detected contacts to the corresponding direct blast. We refer to this operation as contact causality enforcing. In fact, depending on the duty cycle and on the source–target geometry and the ping repetition interval, some contacts generated by the previous ping might appear as received before the direct blast is emitted in the currently processed file. This is explained in more details in Section III-I and Fig. 7. Except for this last component (cluster reordering), which, as will be explained later, is required due to some specific implementation choices, this structure reflects the typical processing chain used for pulsed sonar processing (see, for instance, [12]). Also note that in a general sonar system, contacts can be forwarded to a tracker for track management [17].

Although the tracking component is very important, its description goes beyond the scope of this paper.

The conversion of acoustic data from integer to floating point format is performed using vectorized routines (see Section III-J on code optimization). Acoustic data are usually stored in integer format as produced by an AToD Converter. The integer values are typically stored simultaneously as they are produced by the AToD converter: The data from all the elements of an array at a given instant are acquired and stored; the data from the next sampling interval are stored; and so on. These data are converted and reordered in such a way that data from a given channel are adjacent in the computer memory and channels are stored consecutively with channel number. This reordering allows faster array processing in the subsequent algorithms. Note that, thanks to the usage of a frequency-domain basebanding algorithm described later on in the processing chain (see Section III-D), detrending is not necessary.

B. Initialization Phase

Before starting the actual processing, all the data structures must be initialized. This is done only once at the beginning of the processing chain (or every time a relevant parameter of the CAS algorithm changes during data acquisition), after the first data file is read, and the dimension and the sampling frequency of the input data are obtained. The optimization of this phase is not strictly necessary for the real-time processing; however, since the same library is also used for postprocessing, reducing the initialization time was worthwhile. Most of the improvement was obtained by calculating the beamformer coefficients (see Section III-E) using a public-domain library that vectorizes the calculation of the complex exponential [24].

C. Fourier Transform

Whenever new data are made available for processing, they are transformed into the frequency domain. The transform is done using the fast Fourier transform (FFT) algorithm implemented in the `FFTW` library [25], a well-known public-domain library. The previous CMRE PAS processing only used FFT lengths that were integer powers of two while trading off FFT processing time and the length of overlapping segments [12], whereas the CAS processor relies on FFT lengths that are powers and multiples of two, three, five, or seven. The algorithms that use lengths other than powers of two are marginally less efficient than the classical Cooley–Tukey FFT algorithm [26], but the reduction in the FFT length that can be better adapted to the data length (with less zero padding necessary) offsets the efficiency loss and reduces calculation time in the subsequent steps of the algorithm.

D. Baseband Algorithm

A basebanding algorithm is applied to the data to allow a reduction of the sampling rate, which reduces the computation time of subsequent algorithms. A signal at baseband maintains all the information in the signal band, but is shifted in frequency so that the sampling rate can be reduced to the bandwidth of the (complex) signal rather than twice the highest frequency contained in the original signal [27].

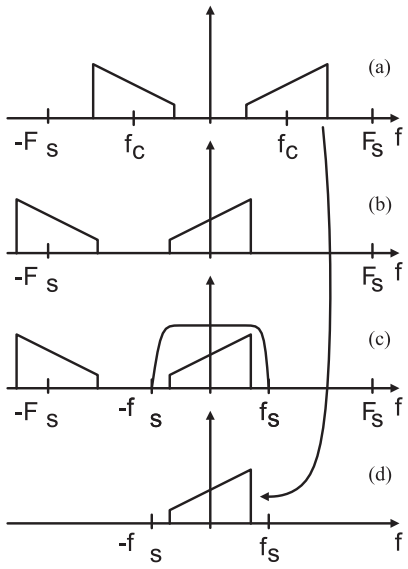

Fig. 5. Schematic depicting the baseband algorithm. (a) Original signal's spectrum. The spectrum is shifted via multiplication by a complex exponential in the time domain, resulting in a complex signal whose spectrum is shown in (b). A time-domain lowpass filter is applied in (c) to remove the lower sideband and any aliasing, resulting in the baseband signal, shown in (d). Alternatively, the operation can be performed directly in the frequency domain by shifting bins.

The time-domain baseband algorithm used in the original PAS software is depicted in Fig. 5. A real signal is acquired at a given sampling frequency F_s. In this algorithm, the signal always remains in the time domain, but for the purpose of visualization, the signal is shown in the frequency domain in Fig. 5. The signal is multiplied by a complex exponential with carrier frequency $-f_c$, which shifts the spectrum down by f_c, as shown in Fig. 5(b). This is followed by a lowpass filter to remove the lower sideband of the spectrum, which is shown contained in the negative side of the spectrum in Fig. 5(c), but could also appear on the positive side due to aliasing. After the lowpass filter is applied, the signal is amplified by a factor of two to regain the energy removed by the filter, resulting in the signal shown in Fig. 5(d). The signal can then be downsampled as long as the resulting sampling frequency is greater or equal to twice the highest frequency contained in the baseband signal f_s.

This time-domain baseband algorithm is efficient for large downsampling factors (larger than 100). For the wideband low-frequency waveforms typical of CAS, the downsampling factor is usually below 10. In this case, frequency-domain baseband algorithms are more efficient; thus, the original time-domain algorithm was replaced with a frequency-domain implementation.

The frequency-domain baseband algorithm developed for the CAS processor can also be visualized using Fig. 5. The first step is applying an FFT to the signal, as shown in Fig. 5(a). The baseband signal in Fig. 5(d) can then be directly obtained by simply shifting the positive frequency samples around zero (amplified by a factor of two) and setting samples in the unwanted band to

zero. This can be done without signal distortions, thanks to the usage of a passband filter with a 90-dB attenuation stopband, which makes the signals band limited. This basebanding algorithm does not need a time-consuming finite-impulse response filter and is, therefore, faster than the time-domain baseband algorithm.

E. Beamforming

A time-domain beamformer operates by delaying the signal recorded on each element of an array and summing the signals from the array elements together. The delay applied to each element is calculated from the angle for which a beam is to be steered. A spatial shading function is also applied to reduce side lobes.

Beamforming can also be performed in the frequency domain, which is the approach used in the PAS software [12]. In this algorithm, the delay operation that would be used in the time domain is replaced by a multiplication of the signal in the frequency domain with a complex exponential.

For CAS processing, the existing beamformer was optimized to exploit vectorization and multithreading. The beamforming algorithm consists of vector multiplication and summation, which are operations that require relatively small efforts to vectorize and program to run over multiple cores.

The frequency-domain beamformer output for frequency ω, steered in the direction θ, is [12]

$$B(\theta,\omega)=\sum_{m=0}^{M-1}X(m,\omega)S(\theta,\omega,m) \tag{1}$$

where $X(m,\omega)$ is the frequency-domain data for the array element m, and $S(\theta,\omega,m)$ are the frequency-domain steering coefficients (or phase shifts)

$$S(\theta,\omega,m)=e^{-j\cos(\theta)\frac{\omega}{c}\Delta_m} \tag{2}$$

where Δ_m is the distance of each array element m from the phase center of the array, ω is the frequency, and c is the speed of sound in water.

Note that the data are kept in the frequency domain after beamforming because the matched filter is also implemented in the frequency domain.

F. Matched Filtering

Fig. 6 depicts how the beamformed data are matched filtered. The matched filter template T, whose spectrum is shown in red, is multiplied by the beamformed data in the frequency domain [see Fig. 6(a)], which is equivalent to correlation in the time domain. Furthermore, a bandpass filter K can be applied via multiplication with the matched filter template. The signal is then band based [see Fig. 6(b)] and transformed into the time domain using the inverse fast Fourier transform (IFFT), resulting in a complex envelope signal represented in blue in Fig. 6(c). The magnitude of the complex envelope is then calculated, shown as the red signal in Fig. 6(c). The envelope operation effectively transforms the signal into baseband, so the complex envelope magnitude squared resembles that depicted in Fig. 6(d).

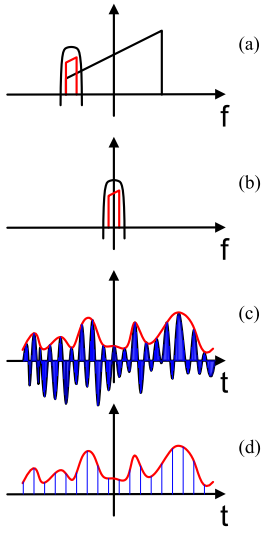

Fig. 6. Schematic depicting how beamformed data are matched filtered and downsampled.

G. Downsampling

The envelope squared of the matched-filtered data can be downsampled by an integer factor n by simply taking every nth sample. This is possible because the matched-filtered signal is limited to the band of the matched filter, which can be reinforced with an additional bandpass filter, as discussed in Section III-F.

The downsampling is depicted in Fig. 6(d), in which the blue vertical lines represent the nth samples being taken, with the rest of the samples being discarded.

H. Normalization and Clustering

After matched filtering and downsampling, the data are in the form of a time series per beam. Normalization is then performed in the time domain along each beam with a median or a split-window normalizer [28].

The median filter implemented, which is based on the approach described in [29], defines a running window that slides through every samples of the file. The output of the filter, given a sample $x[k]$, $k=0,\ldots,N-1$, where N is the length of the sequence, is defined as (3), shown at the bottom of this page,

$$y[k]=\begin{cases}\text{median}(x[0],x[1],\ldots,x[n-1]), & \text{if } k\le(n-1)/2\\ \text{median}(x[N-(n+1)/2],\ldots,x[N-1]), & \text{if } k\ge N-(n+1)/2\\ \text{median}(x[k-(n-1)/2],\ldots,x[k+(n-1)/2]), & \text{otherwise}\end{cases} \tag{3}$$

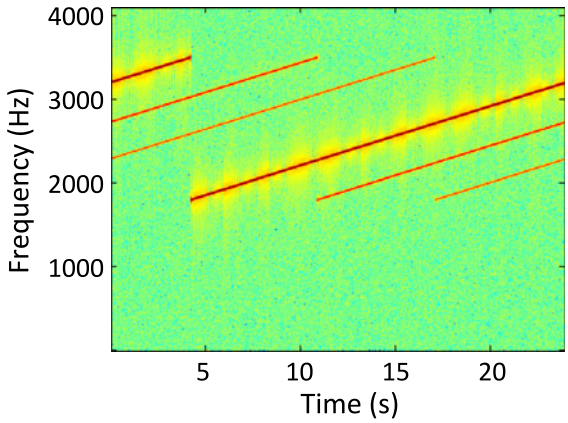

Fig. 7. Spectrogram of a simulated CAS signal as recorded by a single element of a SLITA array as recorded on disk after the acquisition phase. Due to the architecture of the acquisition system, target arrivals might appear as if they are received before the direct blast is transmitted (see arrivals from 2300 to 3200 Hz). These are due to arrivals produced by the previous ping.

where n is odd and corresponds to the window length, and median() is given by

$$\text{median}(x[0], x[1], \ldots, x[n-1]) = X\left[\frac{n-1}{2}\right] \qquad (4)$$

where $X[i]$ is the ascending order sequence obtained from $x[i]$, $i = 0, \ldots, n-1$. Finally, the SNR can be calculated as SNR $= x[k]/y[k]$, for $k = 0, \ldots, N-1$.

The median filter is, in general, more efficient and more robust to large amplitude outliers than the split-window normalizer.

The normalized time series is finally passed on to the detector/classifier, and only the m (user selectable parameter) highest contact level range-bearing points are returned by the processor. With respect to the PAS approach in which the strongest SNR range-bearing point can be associated with the direct blast, in the CAS context this might not always be true, and during our (in the field) operations, we have had echoes with higher SNR than the direct blast. This is because the received acoustic data are split by the acquisition system into files of fixed length, usually coinciding with the PRI. Although the receiver's acquisition system is synchronized with the direct blast transmission time, the time at which the signal is received depends on the specific source–target–receiver (AUV) geometry. For this reason, it appears on each file with a delay from the file start that depends on the source–receiver distance.

When the direct arrival overflows to the next file, the matched filter algorithm becomes less efficient and the SNR of the direct blast decreases (see Fig. 7). The same effect might reduce the target echo as well. No specific algorithms are used to avoid edge effects, and in cases when the direct blast remains below the desired detection threshold, the corresponding subband is simply discarded.

With respect to the PAS algorithm, the detection and clustering algorithms underwent thorough but minor code optimizations, mostly aimed to adapt existing code to the CAS subband processing (e.g., addition of parameters to identify the subbands corresponding to the clusters).

I. *Enforcing Contact Causality*

Given the specific structure of the proposed signal processing chain, one last step must be highlighted. In particular, the detection cluster output is passed through a reordering process before being input to the tracker. This reordering is required because the data are stored in relatively short files, and the processing requires access to information from multiple files at the same time, which was not previously required for PAS processing. The details of this step are now presented.

A CAS system transmits continuously, beginning its next transmission cycle right after the previous one is completed. However, the described signal processing chain focuses on a single SLITA file. This creates the peculiar situation in which target arrivals from the previous transmission may be received before the direct blast of the current transmission is recorded, which we name *noncausal arrivals*.

An example of noncausal arrivals is shown in Fig. 7. The transmitted CAS waveform is a 24-s LFM sweep (1800–3500 Hz), operating at 100% duty cycle, of a 24-s PRI. The direct blast is received at approximately 4 s, with echoes arriving at approximately 11 and 17 s. Since the SLITA file length is also 24 s, the direct blast is cut off by the end of the file; however, it is recorded in the next file. For example, the end of the previous direct blast, sweeping from 3100–3500 Hz, can be observed in the first 4 s of the file shown in Fig. 7. The echoes are also split between files; thus, the first part of each SLITA file contains arrivals that are due to the direct blast of the previous ping.

The echoes need to be associated with the proper direct blast; otherwise, the detection clusters will not be formed properly and this will cause problems for the tracker. A specific process was added to the processing chain to solve this problem. This process reads two consecutive outputs of detection clusters and reorders them to produce a new cluster report with the acoustic detections referenced to the proper direct blast. With this approach, noncausal arrivals are assigned to the contact file of the previous ping; however, this process introduces latency in the system as it is necessary to wait for the next ping to be fully processed before producing detection clusters. Therefore, for a 24-s CAS transmission and 24-s SLITA files, there is a 48-s lag (plus the processing time for one file, which depends on the bandwidth of the transmitted pulse [30]) between the start of the transmission and when the tracker receives new contacts, as opposed to 24 s (plus the processing time) for PAS processing.

J. *Code Optimizations*

Vectorization was employed to increase computational speed. Modern central processing units (CPUs), including those onboard the OEX, use registers. Registers are temporary storage blocks of 128, 256, or 512 b in the most recent architectures. Vectorization is a practice that maximizes the parallelization resulting from the use of large registers. It can increase the computation speed by a factor of 8 for single precision variables and by 4 for double precision variables. Vectorization can be implemented in the C language for simple functions such as summation or multiplication of two arrays. The public-domain

library YEPPP can be used for more complex functions, such as cosine and sine [24].

Multithreading is another way to achieve gains in processing speed. Software can be developed to take advantage of multicore CPUs, which are found in most latest generation computers or single board computers. Multicore CPUs can also be more energy efficient, which is an important factor for AUVs that have limited battery power. It is possible to transform a single-core code into a multicore code with relatively low efforts using standard software development tools [31]. There will be at least four processors in the CPUs on which the CAS processing will run; therefore, parts of the code have been modified to exploit the multithread processing.

IV. Algorithms and Software Implementation in MOOS-IvP

MOOS-IvP is the software infrastructure used onboard the OEX AUVs [21], [23]. MOOS is a publish/subscribe framework that is used for interprocess communication. The core component of MOOS is a central server, called MOOSDB. The MOOSDB acts as a bulletin board that holds the current state of the variables provided by or required by all the onboard signal processing, communication, navigation, and autonomy processes. Processes can subscribe to and publish variables to the MOOSDB to exchange information. By providing this middleware layer of communication, new processes may be added quickly and easily, and may interact with legacy processes or replace them.

The CAS signal processing chain fits within the MOOS architecture, and its implementation within this framework is described in the following.

The front-end interface between MOOS and the signal processing library is called `pProcessCAS`. This interface allows the publication of acoustic detections to the MOOSDB server, thus integrating the CAS processing with the OEX system.

Fig. 8 shows a schematic overview of some of the processes in the MOOS infrastructure. Data recorded for each channel of the SLITA array are saved into a file by the slita-pc and made available to the MOOS system via NFS. Note that no MOOS-based capability is used to handle the array data as MOOS is designed to handle short data transfers. The signal processor, contained in the `pProcessCAS` MOOS process, detects the creation of a new file and reads it into memory. Once the file is loaded into memory, the processor moves through the CAS processing chain described in Section III. The output of the CAS processor is contact reports (i.e., range, bearing, SNR, peak level), which are published into the MOOSDB via the variable `CLUSTER_REPORT`. This variable is read by the `pBistaticLocator` MOOS process for geolocalization using the bistatic equation [32], and can be read by a tracker for track formation and management [33]. Contacts (and tracks) are also sent to the pHelmIvp process that manages AUV behaviors and reconciliate vehicle decisions based on the available information [34]. Examples of AUV behaviors able to exploit contacts and tracks in real time are reported, for instance, in [19] and [35]. In the former paper, the vehicles collectively decide, based on the shared contacts, how to move in the environment to maximize the probability of target detection. In the latter, the behavior implements a nonmyopic control policy to minimize the expected target localization error of the onboard tracking filter.

Contacts and tracks are also acoustically transmitted to the command and control center to be plotted on a screen. The usage of acoustic communications limits the available bandwidth and achievable distances [36], with typical bit rates up to about 900 $\mathrm{b{\cdot}s^{-1}}$ (depending on the network setup and number of nodes) for a 8-km nominal maximum range, requiring an adequate prioritization of the information to be transmitted. For instance, in our experimental setup, the acoustic network had enough available bandwidth to share with the C2 only 5 out of 160 contacts and three tracks [18].

Note that in our CAS scenario, in which the sonar source is continuously moving, towed by the ship, the pBistaticSource process becomes very important. It is the part of the AUV software that is responsible to receive acoustic updates of the source position based on the ship GPS, and to provide position estimation when no updates are received (e.g., due to acoustic packet loss). Since, in our case, the *NRV Alliance* moves at a constant speed along straight lines, this is done by applying a simple constant-velocity kinematic model of the ship, and applying an appropriate position offset to keep into account the distance between the source and the GPS of the ship.

V. Field Trial

The COLLAB–NGAS14 sea trial was conducted with the cooperation of the Italian Navy between October 19 and 31, 2014, off the coast of La Spezia, Italy.

Our primary platform of operations was the *NRV Alliance*, which hosted the command center in the laboratory area, and from where the AUVs were deployed and recovered daily. Additionally, for the CAS operations, the *NRV Alliance* was also deploying the mid-frequency (MF) active source (see Section II-B) and towed an ER target simulator for testing of the multistatic processor. The ER simulated a target by adding a 3-s delay before repeating signals. In this case, the ER did not apply a target response and the signal was transmitted as recorded, using a target strength TS $= 0$ dB. As will be described later, a specific algorithm was used by the ER to downshift a copy of the source transmit signal and to handle the near-continuous waveforms used. Ideally, the ER would have been towed by a second ship to provide a more realistic target simulation. However, this was not possible during COLLAB–NGAS14 and the presence of a distant target was simulated only through the echo repeat delay. With real targets, the scattering is expected to be lower, dependent on the geometry and on the noise, with reflections typically transient and with only momentary opportunities.

A. Site Description

The experimental site is located near 43.8298N, 9.6072E, south-west of the Isola della Palmaria, Italy, and its bathymetry is shown in Fig. 9. The entire area of operation is a 7-km square. The water depth in the area goes from around 50 m (north-eastern part of the area) up to 350 m (south-west region).

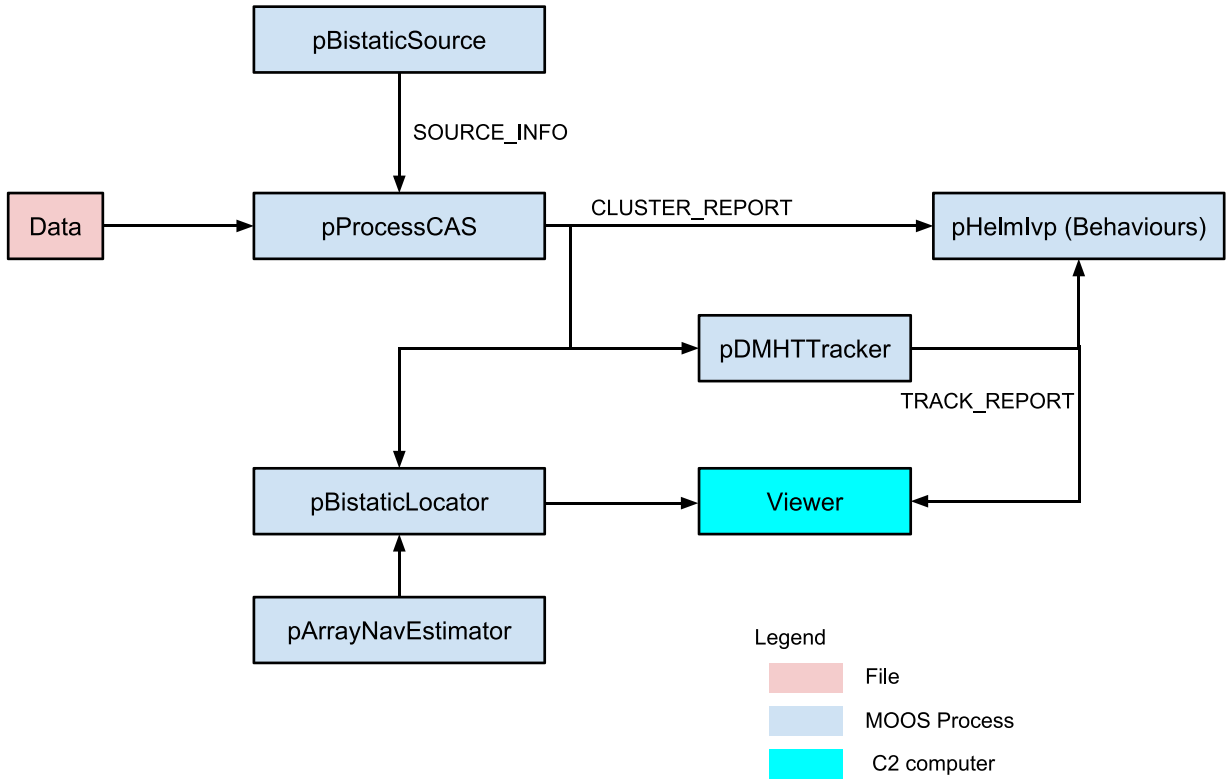

Fig. 8. MOOS-IvP processing scheme onboard the OEX AUVs. After the data file is created (24 s long), it is processed by pProcessCAS. Detections in range and bearing, which will form the final clusters, are passed on via the MOOS variable "CLUSTER_REPORT" to the DMHT tracker for track management and to the AUV behaviors

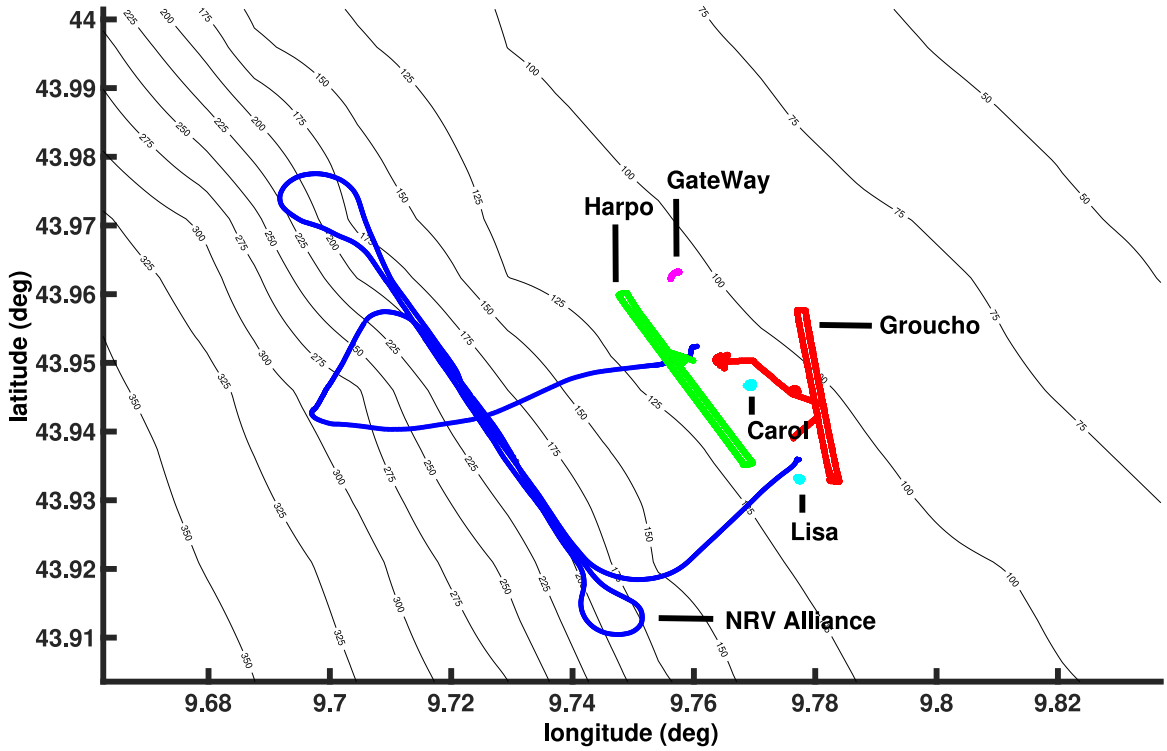

Fig. 9. COLLAB–NGAS14 network for October 21, 2014. OEX Groucho is shown in red, OEX Harpo is shown in green, *NRV Alliance* is shown in blue. Wave Glider Lisa and Wave Glider Carol shown in cyan and coral, respectively, and the Gateway buoy shown in magenta were deployed and used as communication relays and did not run any CAS signal processing. The bathymetric lines are shown with an increase step of 25-m depth.

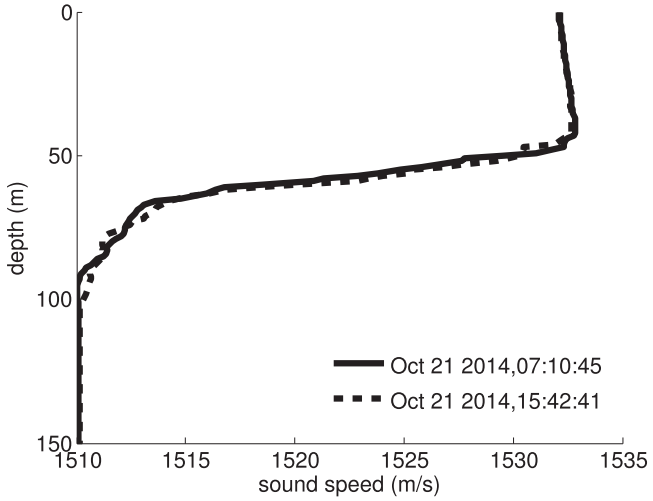

Fig. 10. CTD profiles measured on October 21, 2014.

Two sound-speed profiles, as measured at a location close to the north-east corner of the area on October 21, 2014, are shown in Fig. 10. Note that the thermocline begins at 50 m creating a surface duct and a deep channel.

To give an idea of the acoustic propagation encountered at sea, the transmission loss calculated using the KRAKEN acoustic code [37] is shown in Fig. 11. Note how the signal propagated better for shallow source depth.

B. Experimental Results

This section describes CAS experimental results, obtained using data collected on October 21, 2014.

The MF ATLAS source was deployed and a composite CAS waveform on a 24-s PRI was transmitted. This waveform was comprised of an 18-s duration LFM in the 1800–2500-Hz band synchronized with the GPS clock (direct blast), and a 3-s delayed upshifted replica in the 3000–3700-Hz band for capture, band shifting, and retransmission by the ER. The receive system on the AUVS was also synchronized with the GPS time so that the triggering of the direct blast on the source would trigger the acquisition on the receivers. Since, in this experiment, the source and the ER were both towed by the *NRV Alliance*, the presence of a distant target was simulated tuning the transmission delay of the upshifted replica. Note that in our setup, the maximum duration of the CAS waveform was constrained by the limitation of the ER, which needs 6 s to downshift the received signal and to prepare the demodulated replica for transmission during the subsequent PRI. This leaves a maximum of 18 s for the entire waveform. Since the ER captured signal is out of its transmit band, it could simultaneously record and transmit (see Fig. 12).

Both OEX Groucho and OEX Harpo were deployed into racetracks, as shown in Fig. 9. OEX Groucho was commanded to navigate at the constant depth of 25 m, while OEX Harpo navigated at 35 m. The ATLAS source was deployed at 46.5 m, and the ER was at 40-m depth.

The real-time CAS processing on both the AUVs worked as expected, with each AUV reporting CAS contacts approximately 2.25 km on the far side of the *NRV Alliance* from their position. This was consistent with the 3-s delay in the out-of-band replica. To the best of our knowledge, this was the first successful demonstration of the real-time CAS processing onboard AUVs.

Fig. 12 shows the spectrogram of a single hydrophone (number 8) for a 24-s SLITA data file, acquired on OEX Groucho. The spectrogram shows the 18-s direct blast from 2.7 to 20.7 s sweeping from 1800 to 2500 Hz. The lower level ER signal can also be observed with a delay of 3 s from the direct blast.

The CAS software used eight overlapping subbands, each of which was 4-s long and overlap the previous band for 2 s. The software was set up to produce up to 20 contacts per subband, yielding up to 160 total contacts. The detection threshold was set at 6 dB.

For better visualization, only the ten highest SNR contacts per subband are overlaid on the spectrogram in Fig. 12.

The contacts are placed according to the start frequency of the subband and the time delay of the detection, thus there are eight rows of ten contacts. The red, square-shaped contacts correspond to the direct blast, and the magenta, triangular contacts correspond to the ER. There are eight contacts formed for both the direct and ER arrivals, thus a detection is made in every subband for both cases. This confirms that the software is functioning properly. The remaining green, cross-shaped contacts represent clutter detections (false alarms).

It is finally worth noticing that bottom and surface bounces have been filtered out in processing using a temporal blanking window of tunable length, which was set to 400 m to adapt to the maximum water depth of the area, and hence, they do not appear as contacts in the final output shown in Fig. 12.

To show the CAS performance over time, the beam collapsed plot is shown in Fig. 13. This plot is obtained by selecting the maximum output for each range over a selection of beams (i.e., the beam where the maximum SNR occurs, the preceding and the subsequent beams) and by repeating this procedure for each ping. This kind of detection output is helpful to show the strong echoes obtained during the run. Fig. 13 shows the beam collapsed plot that is obtained by matched filtering each one of the eight subbands, whereas Fig. 14 shows the beam collapsed plot obtained by matched filtering the entire 18-s-long signal. Two main points can be drawn from these results. First, the normalized amplitude output obtained when subbanding is about 5 dB higher than when performing the longer matched filter. This difference is mainly attributed to the Doppler sensitivity of the higher time–bandwidth product obtained when processing the full 18-s-long signal. This can produce dramatic decrease in the SNR if not properly compensated [1]. Also note that in this specific environmental condition, the channel coherence was shorter than 18 s, hence impacting more the SNR on the longer matched filter signal.

Second, the subbanding algorithm, which is able to produce up to one target contact per subband, makes it possible to better exploit the quasi-continuous ensonification of the target. The presence of a target (or of a target-like object) implies a higher degree of correlation of those contacts that are generated by the target (or by a target-like object) between consecutive subbands.

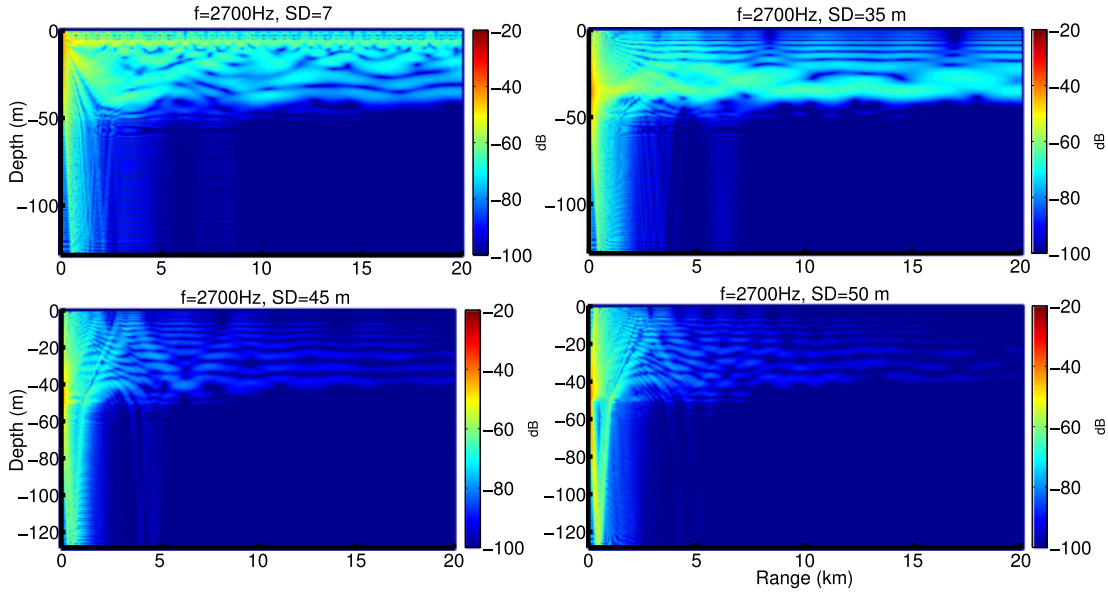

Fig. 11. Transmission loss on October 21, 2014 as calculated using the KRAKEN acoustic code, at various source depths. Note that the ATLAS source was deployed at 46.5 m, the ER at 40 m, OEX Harpo at 35 m, and OEX Groucho at 25 m.

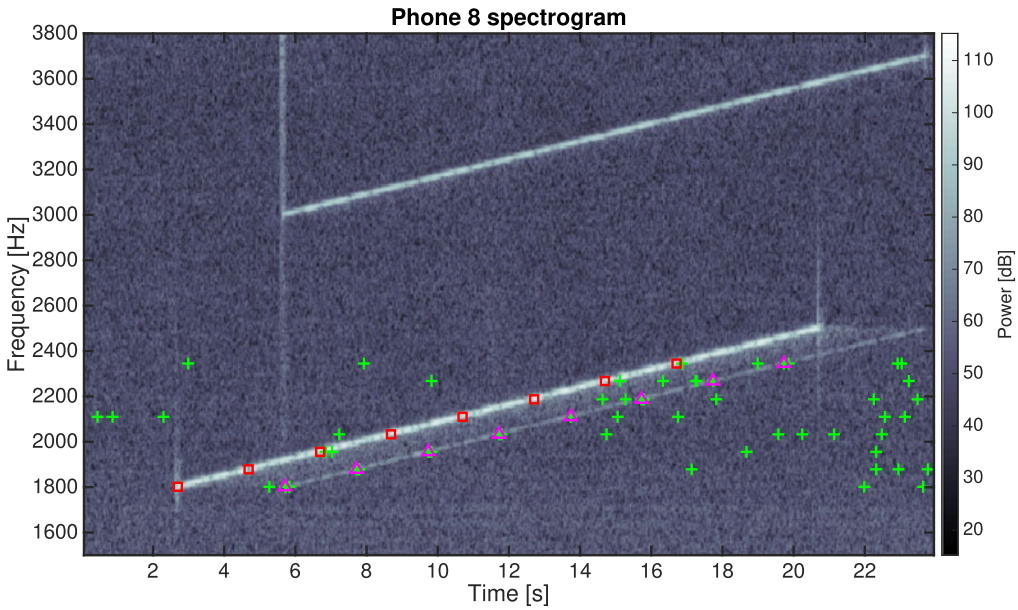

Fig. 12. Spectrogram of a 24-s file containing raw data from a single element of the SLITA array. The overlaid symbols show contacts produced by the processor and mapped back to the raw data spectrogram using the subband start frequency and the contact time delay. The magenta (square-shaped) contacts correspond to the direct blast, the magenta (triangular) contacts correspond to the ER with 3 s delay from the direct blast, and the remaining green (cross-shaped) contacts correspond to false alarms. Note that, for clarity, only the ten highest SNR contacts per subband are shown in the picture. Note the presence of the out-of-band signal that is also transmitted by the active source. This is the part of the signal that, once recorded by the ER, is downshifted to appear as the target echo.

Together with an increase in the number of contacts, the usage of CAS subbanding can help the tracker discard false tracks more quickly.

B. Evaluation of Tracking Performance

The increased tracking performance that can be obtained using CAS subbanding can be shown applying a Bayes filter [38] to estimate the probability of a target being at position $\boldsymbol{x}$, where $\boldsymbol{x} = (r, \theta)$, and r and θ are the detected contact range and bearing. Let us assume a uniform probability of detection P_D in the operating area where a target is assumed to be present. The signal processing algorithm proposed herein always produces N contacts (i.e., measurements $\boldsymbol{z}$), where N is a user-specified parameter. To take this into account, the measurement likelihood function can be constructed as a linear combination of two individual probability density functions (pdfs). The first pdf is modeled as a bivariate, multimodal distribution obtained

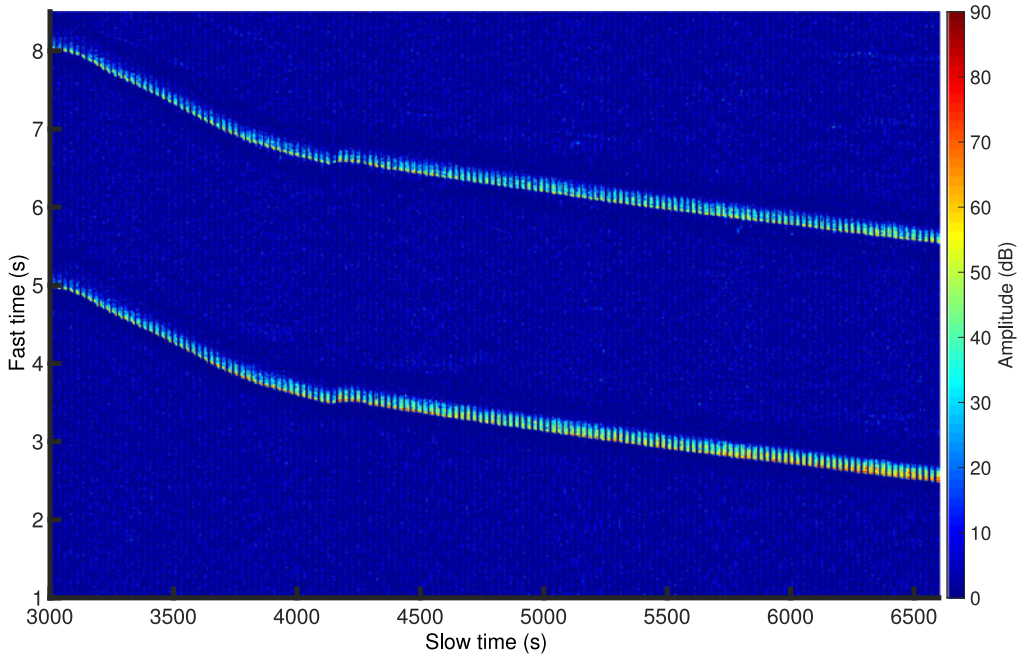

Fig. 13. Beam collapsed plot obtained using eight overlapping subbands of 4 s, with an overlap of 2 s, and displaying per each ping the maximum of the matched filter for each range, after normalization. The horizontal axis is slow time (real time or ping time in seconds from the beginning of the mission), and the vertical axis is echo arrival time in seconds.

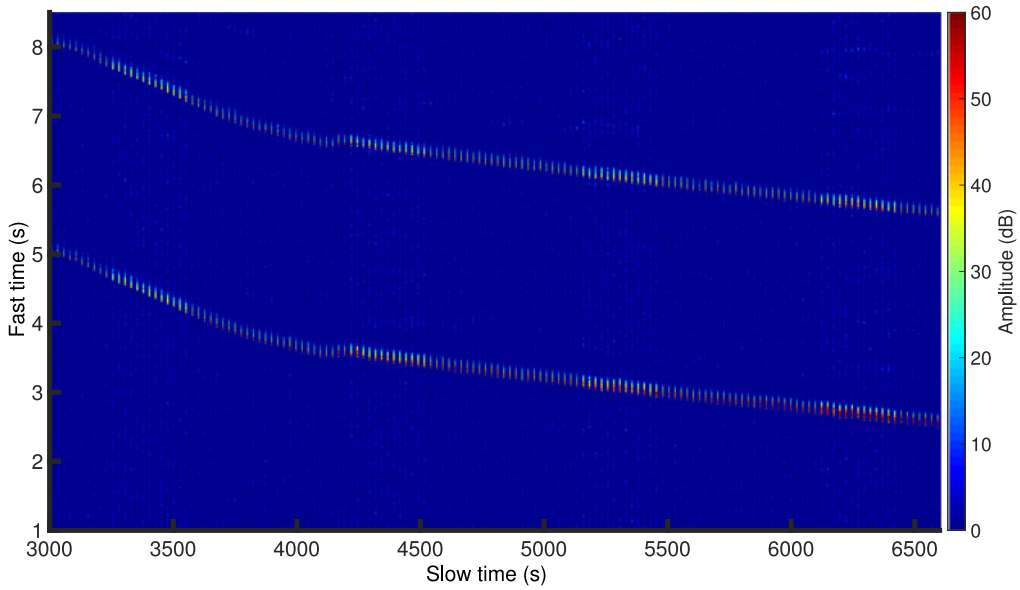

Fig. 14. Beam collapsed plot when matched filtering the CAS signal with only one band of 18 s, and displaying per each ping the maximum of the matched filter for each range, after normalization. The horizontal axis is slow time (real time or ping time in seconds from the beginning of the mission), and the vertical axis is echo arrival time in seconds. Note that the maximum amplitude is 85 dB, but in this picture the dynamic range has been cut to 60 dB to better highlight the contacts produced.

summing up normal distributions with the same variance but different means (i.e., each Gaussian is centered in the associated contact). The second one takes into account that, given that the target can be in any part of the detection region, the contacts produced by the detector could not correspond to a target detection. In other words, even when the detector is not able to detect the target, the algorithm still produces N contacts (false detections). This latter condition is modeled using a uniform distribution spread over the entire area $\mathcal{A}$. The measurement likelihood is hence given as

$$p_m(z|x) = P_D \sum_{i=1}^{N} \gamma \mathcal{N}(r_i, \theta_i, \sigma_r, \sigma_\theta) + \frac{1 - P_D}{\int_{\mathcal{A}} r dr d\theta} \quad (5)$$

where σ_r and σ_θ are the range and bearing standard deviations, which depend on the matched filter applied to the specific CAS signal and on the beamformer of the linear array [39], and γ is the appropriate normalization factor.

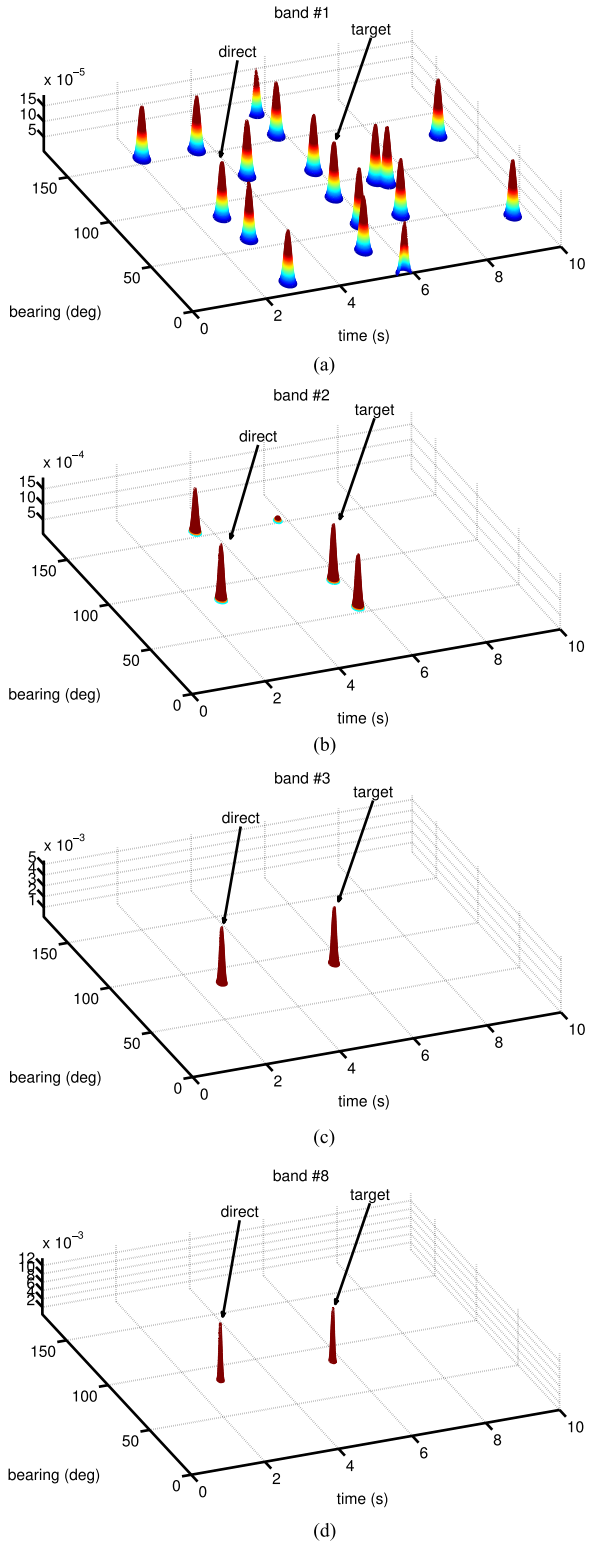

Fig. 15. *A posteriori* pdf of target presence after applying a Bayes filter over the subbands, within a single PRI. Note that, for visual clarity, only the first 10 s out of a 24-s length PRI are shown. The direct blast and echo coherence over the subbands allows them to stand out with respect to the other false contacts that instead quickly fade out over time. (a) pdf after only the first subband is processed. All detections are equally important. (b) pdf when the detections from both the first and the second subbands are used. (c) pdf when detections from the third subband are also used. In this specific case after three subbands, the filter is already able to correctly identify both the direct blast and the target. (d) pdf at the end of the PRI (24 s), after all eight subbands are used.

Let now $p_t^{n-1}(\boldsymbol{x})$ be the pdf giving the probability of a target present at position $\boldsymbol{x}$ at time $n-1$. The target posterior distribution is given as

$$p_t^n(x|z) = \eta p_t^{n-1}(x) p_m(z|x) \tag{6}$$

where η is a normalizing constant.

Fig. 15 shows the result of applying (5) and (6) recursively to the detections obtained in each subband for the data shown in Fig. 12. Since false alarms are uncorrelated between subsequent bands, at the end of the PRI (i.e., after processing the $M = 8$ subbands) their disturbance is filtered out, while the two main contacts, i.e., the direct blast and the received target echo, are positively reinforced. Note how, in this specific case, after only three subbands (i.e., 8 s) the filter is already able to correctly identify the target location. This can be compared to what would happen with a PAS system in which the same information can be gained only after three PRIs, i.e., for the same system settings, 72 s, more than a minute.

VI. Conclusion and Future Development

This paper described the CAS signal processing suite developed at CMRE to run in real time onboard OEX AUVs acting as receiver of a networked autonomous multistatic sonar system. The new CAS processor was based on existing algorithms developed and thoroughly tested for PAS systems at CMRE.

At-sea results of the CAS software have been presented from the COLLAB–NGAS14 sea trial, where it successfully ran onboard the OEX AUVs as well as on desktop computers on *NRV Alliance*. The CAS processor was demonstrated to detect a target (ER towed by *NRV Alliance*). The detected contacts were also passed in real time to the CMRE's tracker.

The CAS processing is efficient and runs approximately three times faster than the original PAS processing software under the same conditions, i.e., when processing the same 24-s file containing data from 64 hydrophones and sampled at 10000.84 Hz [30]. All computations can be completed safely within the PRI as required for real-time processing.

To the best of our knowledge, the CAS processing scheme demonstrated in this paper is the first real-time deployment of CAS using AUVs as receivers of a networked autonomous multistatic sonar.

It is important to highlight, with respect to the experimental results discussed herein, that the acoustic scattering of real targets is expected to be lower and highly dependent on the geometry and the noise. However, one of the advantages of a CAS system is that of creating more detection opportunities, which, if properly exploited at the tracker level, have the potential to increase the overall performance of the system.

Several improvement to the real-time signal processor can be foreseen at this stage. The usage of alternative CAS waveforms to the ones proposed in this paper can lead to an increase in the SNR and signal-to-reverberation ratio. A careful comparison between the CAS and PAS system is also necessary to fully validate its performance in shallower water, where clutter can present challenges to the sonar system.

CAS systems have the potential to increase tracking performance, thanks to the increased ensonification of the target. This was shown in this paper, demonstrating how subbanding can quickly decrease the weight of the false contacts passed on to the tracker and, hence, increase its ability to discard false tracks quicker. Still, the full potential of using the increased number of contacts made available by the CAS processing has yet to be exploited at tracking, classification, and at data fusion level. Dedicated experiments are foreseen in the future to further investigate these aspects.

Finally, the availability of more data on the target opens up scenarios for the development of novel AUV behaviors, which, taking advantage of the newly available data, can increase the vehicles effectiveness and ability to make autonomous decisions.

Acknowledgment

The authors would like to thank everyone involved in the COLLAB–NGAS14 sea trial, including V. Grandi (Engineering Coordinator), G. Parise (working on echo repeater software), everyone in the engineering department, and the captain and the crew of the *NRV Alliance*. They would also like to thank M. Micheli and L. Morlando for their help in porting the PAS code into the CAS processor, and S. Murphy, J. Bates, and G. Ferri for the fruitful discussions.

References

[1] J. R. Bates, P. C. Hines, G. Cenapa, A. Tesei, G. Ferri, and K. D. LePage, " Doppler estimates for large time-bandwidth products using linear frequency modulated active sonar pulses," in *Proc. 4th Underwater Acoust. Conf. Exhib.*, 2017, pp. 169–176.

[2] P. C. Hines, "Experimental comparison of continuous active and pulsed active sonars in littoral waters," in *Proc. 1st Int. Conf. Exhib. Underwater Acoust.*, May 2013, vol. 7, pp. 51–58.

[3] B. T. Hefner and D. Tang, "Guest editorial target and reverberation experiment 2013 (TREX13)—Part I," *IEEE J. Ocean. Eng.*, vol. 42, no. 2, pp. 247–249, Apr. 2017.

[4] B. T. Hefner and D. Tang, "Guest editorial target and reverberation experiment 2013 (TREX13)—Part II," *IEEE J. Ocean. Eng.*, vol. 42, no. 4, pp. 757–758, Oct. 2017.

[5] R. Plate and D. Grimmett, "High duty cycle (HDC) sonar processing interval and bandwidth effects for the TREX'13 dataset," in *Proc. OCEANS Conf.*, 2015, doi: 10.1109/OCEANS-Genova.2015.7271638.

[6] P. C. Hines and S. M. Murphy, "Measurements of signal coherence for high and low duty cycle sonars in a shallow water channel," in *Proc. OCEANS Conf.*, 2015, doi: 10.1109/OCEANS-Genova.2015.7271358.

[7] S. M. Murphy, P. C. Hines, and K. Dunphy, "Classifying continuous active sonar echoes for target recognition," in *Proc. Int. Conf. Exhib. Underwater Acoust.*, 2014, pp. 811–818. [Online]. Available: http://www.uaconferences.org/docs/Past_Proceedings/UACE2014_Proceedings.pdf.

[8] G. Grimmett, "Target AOU growth containment using LFM high duty cycle sonar," in *Proc. Int. Conf. Exhib. Underwater Acoust.*, 2014, pp. 839–846. [Online]. Available: http://www.uaconferences.org/docs/Past_Proceedings/UACE2014_Proceedings.pdf

[9] Y. Wang and J. Yang, "Continuous transmission frequency modulation detection under variable sonar-target speed conditions," *Sensors*, vol. 13, no. 3, pp. 3549–3567, 2013.

[10] L. Gerken, *ASW Versus Submarine Technology Battle*. Chula Vista, CA, USA: Amer. Sci. Corp., 1986.

[11] S. Lemon, "Towed-array history, 1917–2003," *IEEE J. Ocean. Eng.*, vol. 29, no. 2, pp. 365–373, Apr. 2004.

[12] M. J. Hamilton, S. Kemna, and D. T. Hughes, "Antisubmarine warfare applications for autonomous underwater vehicles: The GLINT09 field trial results," *J. Field Robot.*, vol. 27, pp. 890–902, 2010.

[13] S. Kemna, M. J. Hamilton, D. T. Hughes, and K. D. LePage, "Adaptive autonomous underwater vehicles for littoral surveillance," *Intell. Serv. Robot.*, vol. 4, pp. 245–258, 2011.

[14] G. Ferri, A. Munafò, R. Goldhahn, and K. LePage, "Results from COLLAB13 sea trial on tracking underwater targets with AUVs in bistatic sonar scenarios," in *Proc. OCEANS Conf.*, 2014, doi: 10.1109/OCEANS.2014.7003069.

[15] A. Caiti, V. Calabro, A. Munafò, G. Dini, and A. LoDuca, "Mobile underwater sensor networks for protection and security: Field experience at the UAN11 experiment," *J. Field Robot.*, vol. 30, no. 2, pp. 237–253, 2013.

[16] A. Caiti, V. Calabro, G. Dini, A. LoDuca, and A. Munafò, "Secure cooperation of autonomous mobile sensors using an underwater acoustic network," *Sensors*, vol. 12, no. 2, pp. 1967–1989, 2013.

[17] P. Braca, R. Goldhahn, G. Ferri, and K. LePage, "Distributed information fusion in multistatic sensor networks for underwater surveillance," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4003–4014, Jun. 2016.

[18] G. Ferri *et al.*, "Cooperative robotic networks for underwater surveillance: An overview," *IET Radar, Sonar Navig.*, vol. 11, no. 12, pp. 1740–1761, Dec. 2017.

[19] A. Munafo, G. Ferri, K. LePage, and R. Goldhahn, "AUV active perception: Exploiting the water column," in *Proc. OCEANS Conf.*, 2017, pp. 1–8.

[20] A. Maguer, R. Dymond, M. Mazzi, S. Biagini, S. Fioravanti, and P. Guerrini, "SLITA: A new slim towed array for AUV applications," in *Proc. Acoust. Paris*, 2008, pp. 323–328.

[21] M. R. Benjamin, P. M. Newman, J. J. Leonard, P. Newman, and H. Schmidt, "Extending a MOOS-IvP autonomy system and users guide to the IvP-Build toolbox," Tech. Rep. MIT-CSAIL-TR-2009-037, 2009.

[22] P. Newman, "Moos—A mission oriented operating suite," MIT Dept. Ocean Eng., Cambridge, MA, USA, Tech. Rep. OE2003–07, 2003.

[23] M. R. Benjamin, H. Schmidt, P. M. Newman, and J. J. Leonard, "Nested autonomy for unmanned marine vehicles with MOOS-IvP," *J. Field Robot.*, vol. 27, no. 6, pp. 834–875, 2010. [Online]. Available: http://dx.doi.org/10.1002/rob.20370

[24] M. Dukhan, "Yeppp! library documentation for C programmers," 2013. [Online]. Available: http://www.yeppp.info/c/

[25] M. Frigo and S. G. Johnson, "The design and implementation of FFTW3," *Proc. IEEE*, vol. 93, no. 2, pp. 216–231, Feb. 2005. [Online]. Available: http://http://www.fftw.org

[26] J. W. Cooley and J. W. Tukey, "An algorithm for machine calculation of complex fourier series," *Math. Comput.*, vol. 19, no. 90, pp. 297–301, 1965.

[27] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1975.

[28] A. Baldacci and G. Haralabus, "Signal processing for an active sonar system suitable for advanced sensor technology applications and environmental adaptation schemes," in *Proc. 14th Eur. Signal Process. Conf.* Florence, Italy, Sep. 4–8, 2006. [Online]. Available: https://www.eurasip.org/Proceedings/Eusipco/Eusipco2006/papers/1568981421.pdf

[29] S. D. Mohanty, "Efficient algorithm for computing a running median," Max Planck Institut Gravitationsphysik, Golm, Germany, Tech. Rep. LIGO-T-030168-00-D, Dec. 2003.

[30] G. Canepa, A. Munafò, M. Micheli, S. Murphy, and L. Morlando, "Real-time continuous active sonar processing," in Proc. *OCEANS Conf.*, 2015, doi: 10.1109/OCEANS-Genova.2015.7271658.

[31] "The OpenMP API specification for parallel programming," [Online]. Available: http://www.openmp.org. Accessed on: Jan. 29, 2018.

[32] S. Coraluppi, "Multistatic sonar localization," *IEEE J. Ocean. Eng.*, vol. 31, no. 4, pp. 964–974, Oct. 2006. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4089054

[33] C. Carthel and S. Coraluppi, "The NURC distributed multi- hypothesis tracker (DMHT): Scientific user documentation," , NATO STO Centre Maritime Res. Experimentation, Tech. Rep. NURC-FR-2007-025, Dec. 2007.

[34] G. Ferri, A. Munafò, R. Goldhahn, and K. Lepage, "A non-myopic, receding horizon control strategy for an AUV to track underwater targets in bistatic sonar scenarios," in *Proc. 53rd IEEE Conf. Decision Control*, 2014, pp. 5352–5358.

[35] G. Ferri, A. Munafò, and K. LePage, "An autonomous underwater vehicle data-driven control strategy for target tracking," *IEEE J. Ocean. Eng.*, vol. 33, no. 2, pp. 323–343, Apr. 2018.

[36] A. Caiti, E. Crisostomi, and A. Munafo, *Physical Characterization of Acoustic Communication Channel Properties in Underwater Mobile Sensor Networks*. Berlin, Germany: Springer-Verlag, 2010.

[37] M. Porter, "The kraken normal mode program," SACLANT Undersea Res. Centre, La Spezia, Italy, Tech. Rep. Memo. SM-245, 1991, ch. 2.

[38] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Bradford, U.K.: YBS Publishing, 2011.
[39] R. J. Urik, *Principles of Underwater Sound*. Los Atlos, CA, USA: Peninsula, 1983.

Andrea Munafò (M'09) received the B.Sc. degree in computer science engineering, the M.Sc. degree in automation engineering, and the Ph.D. degree in automation, robotics, and bioengineering, all from the University of Pisa, Pisa, Italy, in 2002, 2005, and 2009, respectively.

From 2009 to 2013, he was a Postdoctoral Research Assistant with the Interuniversity Centre of Integrated Systems for the Marine Environment, Genova, Italy. From 2013 to 2016, he was a Research Scientist with the NATO STO Centre for Maritime Research and Experimentation, La Spezia, Italy, working in the field of cooperative underwater robotics and underwater communications. Since 2017, he has been a Senior Scientist and Engineer with the Marine Autonomous and Robotics System Department, National Oceanography Centre, Southampton, U.K., where he is leading the development of novel onboard autonomy solutions for marine vehicles. His research interests include robotics, adaptive planning and sampling, underwater acoustics, and sonar systems.

Gaetano Canepa was born in La Spezia, Italy, in 1965. He received the Laurea degree in electronic engineering from the University of Pisa, Pisa, Italy, in 1990, and the Ph.D. degree in anthropomorphic robotics from the University of Genoa, Genoa, Italy, in 1994.

He was a Consultant at the University of Pisa and a Research Assistant at McGill University, Montreal, QC, Canada. Since 1996, he has been with the NATO STO Centre for Maritime Research and Experimentation (formerly SACLANT Undersea Research Centre and NATO Undersea Research Centre), La Spezia, Italy, where he has been working on underwater acoustics, sonar modeling and simulation, data and image processing, and real-time algorithm implementation on autonomous underwater vehicles.

Kevin D. LePage received the B.S. degree in naval architecture and marine engineering from Webb Institute, Glen Cove, NY, USA, in 1983, and the M.S. and Ph.D. degrees in ocean engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1987 and 1992, respectively.

He is a Principal Scientist and a Program Manager for the Cooperative Anti-Submarine Warfare, NATO Science and Technology Organization Centre for Maritime Research and Experimentation, La Spezia, Italy, where his work is directed toward capturing requirements for future unmanned solutions for ASW capability under funding from NATO Allied Command Transformation. While with CMRE, he has been a Scientist-in-Charge of five scientific sea trails and the CMRE component of three NATO naval ASW exercises where unmanned systems were deployed and tested.

Dr. LePage was the recipient of the NATO STO Scientific Achievement Award in 2017 for the Development and Demonstration of Networked Autonomous ASW as a Team Leader. He is a Fellow and the Past Chair of the Underwater Acoustics Technical Committee of the Acoustical Society of America.

Document Data Sheet

Security Classification		Project No.
Document Serial No.	**Date of Issue**	**Total Pages**
CMRE-PR-2019-051	May 2019	15 pp.

Author(s)

Andrea Munafò, Gaetano Canepa, Kevin D. LePage

Title

Continuous active sonars for littoral undersea surveillance

Abstract

Recent advances in transducer and computing technology have pushed the concept of continuous active sonar (CAS) or high duty cycle sonar as an area of interest for application to antisubmarine warfare. Unlike conventional pulsed active sonars, CAS processing aims at detecting echoes while transmitting with a nearly 100% duty cycle. This paper describes the signal processing chain developed at the Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy, for real-time CAS processing within an autonomous networked multistatic sonar system. The algorithm uses subband processing, which can potentially provide a higher target update rate than the traditional pulsed active sonar, while maintaining the same search radius. The higher rate of contact information can improve target tracking performance. Performance results are given from the COLLAB–NGAS14 sea trial where the CAS processor was deployed on two autonomous underwater vehicles (AUVs) acting as receivers of the CMRE experimental multistatic demonstrator. Results show the feasibility of the CAS concept in littoral scenarios, using AUVs as real-time receivers.

Keywords

Antisubmarine warfare (ASW), autonomous underwater vehicles (AUVs), continuous active sonars (CAS), multistatic sonars, signal processing

Issuing Organization

NATO Science and Technology Organization
Centre for Maritime Research and Experimentation

Viale San Bartolomeo 400, 19126 La Spezia, Italy

[From N. America:
STO CMRE
Unit 31318, Box 19, APO AE 09613-1318]

Tel: +39 0187 527 361
Fax:+39 0187 527 700

E-mail: library@cmre.nato.int