Multi-everything Sonar Simulator (MESS)

B. La Cour, C. Collins, and J. Landry
Applied Research Laboratories
The University of Texas at Austin
Austin, TX, U.S.A.
blacour@arlut.utexas.edu

Abstract - ***This paper describes the Multi-everything Sonar Simulator (MESS), a system for performing general active or passive undersea sonar simulations. It is designed to provide element or beam-level data, in the time or frequency domain, in either a real or basebanded format. It is capable of handling any number and variety of sources, receivers, and targets with arbitrary, rigid-body trajectories in three dimensions. In general, simulated data will contain contributions from all sound sources: ambient noise, reverberation, source direct blasts, target echoes, target radiated noise and receiver self noise. The underlying signal generation, acoustic propagation, and physical interactions are performed using the Sonar Simulation Toolset (SST), which in turn uses the Comprehensive Acoustic Sonar System (CASS) and Gaussian Ray Bundle (GRAB) algorithm for eigenray generation. Acoustic propagation is ray-based only but is range dependent. The system may be used for generating active, passive or combined active/passive scenarios. It may also be used to inject active or passive targets into existing, real-world data sets.***

Keywords: multisensor, acoustic simulation, active sonar, passive sonar

1 Introduction

Simulation plays a variety of roles in the undersea sonar community. It may serve to help develop and validate new algorithms or test and evaluate complete systems. It may be specific to a particular sonar system, completely general, or anything in between. Simulated data has the obvious advantage of providing absolute ground truth (at least, so far as the inputs are known) and controlled environmental effects. Of course, it may suffer from a certain lack of realism in providing results that are "too clean" and fail to capture important but difficult-to-model real-world effects, such as discrete active clutter or passive transient noise.

To achieve greater realism, it may be desirable to inject simulated active targets or passive sound sources into previously measured, at-sea data. In this manner, realism in the environment (in particular, the presence of false targets) is maintained along with one or more well-controlled, albeit synthetic, targets. Such an approach is limited, however, by the specific scenario (sources, receivers, waveforms, etc.) and environment in which the data was collected. As always, the appropriate tradeoff between realism and flexibility will dictate which approach is best suited to the task at hand.

Innumerable sonar simulators have been developed in the past, each with their varying degree of generality, realism, and ease of use [1]. The variety is indicative of the varied purposes: algorithm validation, tracker evaluation, system performance prediction, and so on. Prechner and Bowen [2] have argued for designing simulators which "provide a flexible framework within which users may model a wide range of sonar types operating in different scenarios," and the design philosophy behind MESS has been very much in keeping with this point of view.

We should also differentiate simulators which are designed to provide data at the level of the sonar equation (transmission loss, reverberation level, signal excess, etc.), such as the Generic Sonar Model (GSM) [3] or its more capable successor, the Comprehensive Acoustic Sonar System (CASS) [4] and related Gaussian Ray Bundle (GRAB) eigenray model [5], with those designed to produce true at-the-sensor element-level or beam-level data. An important example of the latter kind is the Sonar Simulation Toolset (SST) [6, 7], which MESS relies heavily upon. Indeed, MESS may be viewed as an interface to SST, which is itself an interface to CASS/GRAB. SST will be described in more detail in Sec. 2.

The Organisation for Applied Scientific Research (TNO) has, through its Physics and Electronics Laboratory (FEL), developed a sonar simulation system similar to MESS called SIMONA (SIMulation Of Non-acoustic and Acoustic data) [8]. Like MESS, SIMONA provides at-the-sensor data for both passive and active scenarios which is either of a purely simulated or target-injected nature. SIMONA is a well-developed and comprehensive simulation system, but it uses a simple propagation model which assumes a range-independent, isovelocity environment.

The outline of this paper is as follows. An overview of SST capabilities and limitations is given in Sec. 2. In Sec. 3 we described the basic architecture of MESS: what it uses as inputs, how it produces simulated data, and the format of the output. Both full simulations and target injection will be discussed. We end in Sec. 4 with an example, derived from the DEMUS'04 sea-trial, of multitarget injection in a multistatic setting. A summary follows in Sec. 5.

2 Acoustic Modeling with SST

The Sonar Simulation Toolset (SST) is a program designed to create simulated sonar signals which reflect the properties of a given set of sound sources, reflectors, and receivers, with propagation through a fairly realistic, user-defined underwater environment. As such, SST provides as its basic output the calibrated, element-level (or beam-level) signal, in the time or frequency domain, which would actually be recorded at the receiver. Individual signal components, such the direct blast of an active source, a target echo, reverberation, ambient noise, target radiated noise, or receiver self-noise, may be computed individually and added coherently to produce the total received signal. What follows is a brief description of SST from the point of view of MESS. A more detailed account may be found in [7].

SST is written in C++, and the interface belies its object-oriented structure. For the user, SST is a scripting language, much like GSM or CASS. Plain text scripts provide the information needed to define underlying C++ objects, and these scripts are processed in serial fashion. Output, when needed, may be written out to external files as either plain text or in a variety of binary formats. MESS operates by generating the necessary SST scripts, running them, and then collating the resulting outputs.

Acoustic propagation in SST is, at present, solely ray-based. Eigenrays are used to construct a Finite Impulse Response (FIR) filter to describe signal propagation. In a simplistic environment in which the bottom is flat and the sound speed is constant, an analytic model of the eigenrays may be used. If the bottom is flat (i.e., range-independent) but the sound speed varies in depth, GSM may be called to compute the necessary eigenrays. If there is varied bathymetry, in addition to a depth-dependent sound speed, then CASS must be used instead. In addition to bathymetry and sound speed, the environment may be specified by surface, bottom, and volume scattering properties, volume attenuation, and wind speed (for surface reverberation). The bottom properties, in particular, may be spatially varying and can be parameterized by mean grain size [9].

Receivers are defined by their 3-D rigid body kinematic state (position, velocity, orientation, and orientation rate at a given time) and array geometry. A receiver may be defined as a collection of omnidirectional hydrophones, or as an array of elements with varied directivity.

Sources are defined similarly, differing only in that a transmitted signal is specified. The transmitted signal, as well as the received signal, may be described in terms of a time- or frequency-domain representation and may be either a real or complex, basedbanded signal. A source may be used to describe both an active source and a radiating target.

Targets may be modeled with arbitrary complexity. Within SST, one may define a target most generally as a composite of point scatterers, each with a position relative to the body coordinates of the target, an artificial time delay between ensonification and retransmission, and a complex impulse response. The latter, however, is independent of source or receiver position. To obtain a more general highlight response, SST may be used to call an external program, specified by the user, which provides the desired highlight response upon being provided a set of source/receiver positions and a relevant set of frequencies by SST during runtime. In this way, targets of arbitrary complexity may be modeled within the SST framework.

For a passive signal, active direct blast, or target echo, the effects of source, receiver, and target motion (both translational and rotational) are taken into account automatically by SST and are reflected in the Doppler shift of the received signal. This is accomplished by computing several sets of eigenrays and using a convergence algorithm based on Newton's method to locate the receiver at the time of ensonification.

3 Simulation Architecture

To define a simulation run in MESS, one must specify the ocean environment (assumed fixed), and all relevant information regarding the sources, receivers, and targets. This information is used to generate plain text scripts, which are used to run SST. The resulting output is combined coherently to form a set of "scans" describing received signals. Currently, MESS is implemented as a set of MATLAB scripts which drive SST and collate the resulting output.

MESS requires the user to specify a set of parameters describing the unique ocean environment for a particular simulation. By default, MESS utilizes the APL/UW High-Frequency Environmental Acoustic Models [9] for monostatic surface and bottom models to describe attributes of the ocean's bottom and surface. The bottom sediment can be specified to closely model the location of the chosen simulation and may, if desired, be made spatially dependent. It is necessary for the user to define the wind speed, ocean depth or bathymetry, and information regarding the reverberation scattering strengths. For the reverberation surface, bottom, or volume scattering layer the user must specify parameters such as the minimum and maximum depth limits and the scattering strength. Given that the sound speed is a function of depth, the user can input a table of sound speed versus depth or simply provide a constant value for this parameter.

Following the initialization of the ocean environment, physical specifications are made describing the geometry of the sources, receivers, and targets. Within the common data structure that will be discussed shortly, information is created or imported defining the position of each transducer relative to the center of the given source or receiver array.

The common data structure which populates the parameters within MESS contains three main fields where information regarding the receivers, sources, and targets, respectively, is stored. The receivers, sources and targets all have three similar subfields which store the name, trajectory data, and information regarding the noise, if any, generated by the object. Trajectory information includes the time, position, velocity, orientation, and angular velocity over the duration of a given simulation. The times at which the kinematic state is specified are arbitrary, as interpolation is used to determine the state at any particular time of interest. The noise field allows the user to specify a narrowband frequency with the corresponding decibel levels of its harmonics as well as a broadband frequency with an associated spectrum.

Receiver and source fields each contain an element subfield to store information defining characteristics about their transducer arrays. Coordinates of the individual transducers are given relative to the center of the receiver or source array. The directivity and gain are also specified for each array element in their respective subfields.

The receiver field contains a "scan" subfield to store information regarding a particular segment of data. The time, kinematic state, and receiver settings are specified for each individual scan. The kinematic state at a particular scan simply references the trajectory subfield of the receiver and interpolates at the start time of the scan. Settings include the data format ("real" or "complex"), domain ("time" or "frequency"), baseband frequency, start and stop times of the scan, the number of samples taken during the scan, and the sample rate.

Within the source field of the data structure there is a subfield storing all relevant information regarding each source ping. A ping from the source will be comprised of one or more subpulses with associated time offsets given relative to the start of the ping. This level of the data structure is where settings and signal information of each subpulse are stored. Within the settings subfield the user can specify the time offset, center frequency, waveform type, pulse length, and bandwidth of each subpulse. Possible waveforms are Continuous Wave (CW), Linear Frequency Modulated (LFM), and Hyperbolic Frequency Modulated (HFM); others may easily be included.

The user has several options for means of populating the common data structure described above. Initially, the data structure was designed to be populated by simply modifying the parameters within the MATLAB files by hand. The user now has the option of importing data from an actual data set, allowing an artificial target to be injected for the simulation. There are future plans to develop a graphical user interface (GUI) allowing the user to specify information within the data structure more efficiently. The GUI will initially prompt the user to enter the number of sources, receivers, and targets and their relative trajectory information. The user would also be presented with options to import data, and modify existing default parameters.

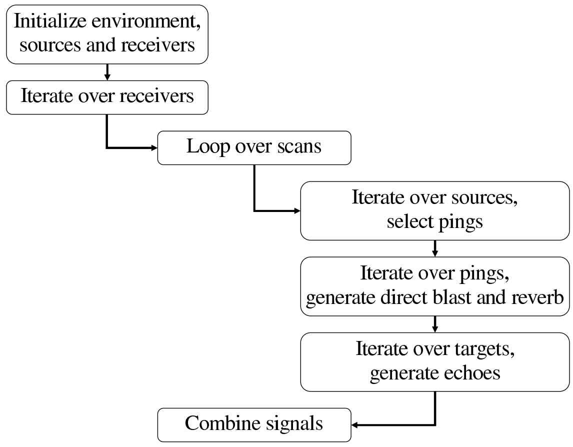

Figure 1: Flow chart schematic of MESS processing.

A process flow diagram is shown in Fig. 1. MESS starts out by writing plain text scripts containing information about the ocean's environment and source and receiver geometries, all of which are assumed to remain constant over the simulation. These SST scripts are later called within other SST scripts during the MESS run. After this step, MESS loops over the number of receivers present during the simulation. Within this loop, MESS loops over each scan for a given receiver. Plain text scripts are generated and then run by SST, giving the trajectory information and settings of the receiver during a scan. For each scan, radiated noise from each receiver, source, and target is determined and propagated to the given receiver. For active signals, MESS finds the sources that are pinging within a given scan or have pinged within a certain length of time (currently specified by the user) before the scan. MESS then loops over these source pings to calculate the direct blast of each. SST is called within MESS for each source ping during this step. Depending on the number of sources pinging in a given scan, SST will return output for each individual direct blast. Similarly, when calculating echoes present SST is called for each target and source present during a given scan. Again, depending on the number of sources and targets present, SST can generate multiple output files for the echoes heard by the current receiver during the scan. Following this step, SST input files are generated which describe the reverberation information, and SST is called another time to generate this reverberation. The final step during a given scan is to combine all of the output generated by SST to generate a single output file for a given scan. This procedure is performed for each scan of the receiver and then repeated for each additional receiver.

4 Example of Target Injection

In this section we illustrate an example using MESS for target injection. The data set used was collected during the DEMUS'04 seatrial, which was performed in

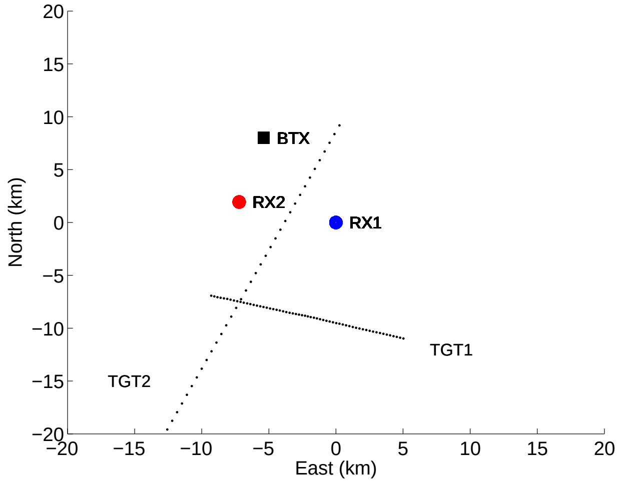

Figure 2: Plot of source, receiver, and target geometry. TGT1 moves slowly from west to east, while TGT2 moves rapidly from south to north.

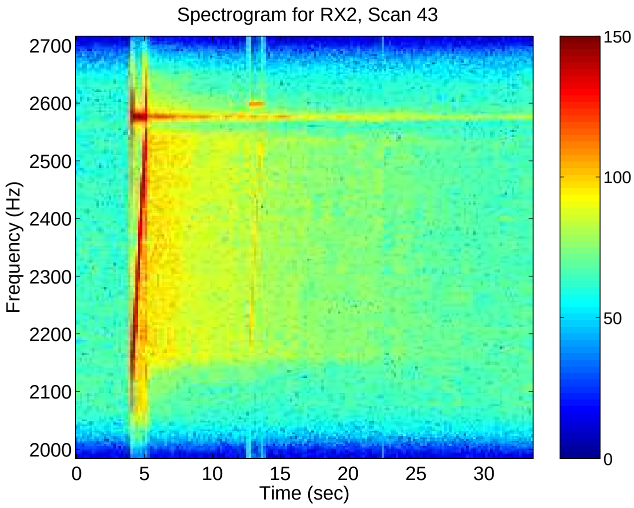

Figure 3: Spectrogram of target-injected data. The color scale is in dB re 1 μPa.

the Malta Plateau region during September of 2004. DEMUS'04 was performed as part of the Deployable Multistatic Sonar Joint Research Project (JRP) formed between the NATO Undersea Research Centre (NURC), the U.S. Office of Naval Research (ONR), and the U.K. Ministry of Defense (MOD). The seatrial involved the use of, among other assets, the Deployable Multistatic Undersea Surveillance (DEMUS) system, a set of moored active sonar sources and receivers.

For the purpose of this example, three components are of interest: one DEMUS source, denoted BTX, and two DEMUS receivers, denoted RX1 and RX2. Each was moored to the bottom at a depth of about 50 m in an area with some 100 m of water depth. (See Fig. 2.) For this particular data set, denoted E06, the DEMUS source pinged once every two minutes with simultaneous LFM and CW pulses of 1-sec duration each. The CW pulse was centered at 2575 Hz, while the LFM pulse was centered at 2350 Hz with a bandwidth of 400 Hz. A total of 60 scans, each about 34 sec long, were produced over a two-hour interval. A single echo repeater was towed at low speeds along a linear trajectory south of the receivers, but in this particular data set the echo repeater was not operational. Thus, no targets were present in the original data.

In the simulation presented here, we used MESS to inject two targets into the E06 data set. The first, TGT1, is a slow-moving target following the trajectory of the original towing ship from west to east at about 4.2 knots. The second, TGT2, is a fast-moving target that enters the sensor region from the southwest, passing between the two receivers in its approach at a constant speed of 14 knots. Both targets were modeled as finite cylinders of length 74 m and diameter 10 m. (See [10], pg. 303, table 9.1, and note that $\cos^2\theta$ should be replaced by $|\cos\theta|$.) The environment was treated as a flat-bottom ocean with a 1500 m/s isovelocity sound speed profile and a silty clay bottom sediment (mean grain size of 8.0).

Processing begins by populating the aforementioned general data structure with kinematic information for the source, target, and receiver, as well as receiver and transmission settings for each component. To correctly match the original data set, all source and receiver information is taken from the original data set and supporting information. This data structure is then passed to MESS, which returns the element-level echo data. (For target injection, of course, MESS does not compute the direct blast or reverberation.) The simulated data is scaled for calibration and then added coherently to the raw data from the original exercise. It is then re-converted to its native binary format. At this point, the target-injected data can be used in exactly the same manner as the original data.

Figure 3 shows a spectrogram plot of the target-injected data on one element of RX2 for scan 43. The direct blast, present in the original data, is clearly visible at approximately 4 seconds. The broadband ringing at the beginning and end of the pulse is due to the square shape of the envelope and appears in the simulated echoes as well. The echo for TGT2 appears at around 12.5 seconds, with a Doppler shift of approximately +20 Hz clearly visible in the CW pulse. TGT1 is far weaker, appearing at approximately 22 seconds with a barely discernible Doppler shift of -5 Hz. TGT2 is also much louder in this scan, as it is nearly at specular aspect with respect to RX2 and is much closer to the source and receiver than TGT1. The time delays and Doppler shifts are in good agreement with predicted values based on source/receiver positions and target kinematics.

Figure 4 displays the SNR of both targets as a function of ping number. The dashed vertical lines indicate where a specular return is expected to occur based on the source/receiver geometry and the target trajectory. It is clearly visible that in most cases, the peak return does occur at or near this ping. It should also be pointed out, that this plot does not take into account distance from the source and receiver, which would explain the relatively low returns off of Target 1 nearer the end, and Target 2 near the beginning of the exer-

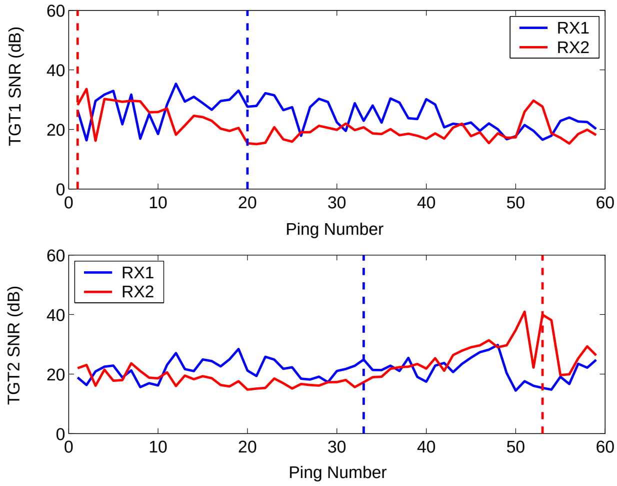

Figure 4: Plot of target SNR as a function of ping number. The predicted ping for a specular return in indicated by the vertical dashed line, with blue for RX1 and red for RX2. The upper subplot is for TGT1 (the slow-moving target), while the lower subplot is for TGT2 (the fast-moving target).

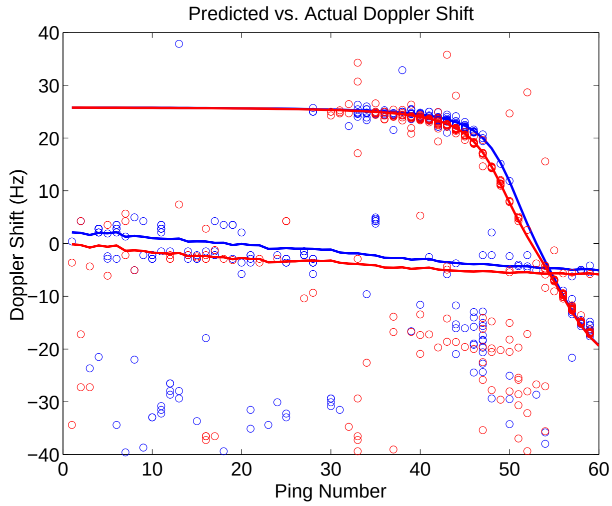

Figure 5: Plot of Doppler shift versus ping number. The solid lines indicate the predicted values for TGT1 (lower curve) and TGT2 (upper curve) with respect to RX1 (blue) and RX2 (red). The circles indicate the measured Doppler shifts for contacts on RX1 (blue) and RX2 (red).

cise.

Fig. 5 shows Doppler shifts estimated from the clutter and simulated targets as a function of ping number. We note that there is a gap where almost no returns off of TGT2 are visible on RX1. This can be attributed to the target entering the blanking region of RX1, where the one-second-duration direct blast occurs. Additionally, there is a large amount clutter present at large negative Doppler shifts. This can mostly be attributed to the LFM sweep also present in the source transmission, whose frequency band reaches up just 25 Hz (15 knots) below the CW center frequency.

5 Summary

In this paper, we have described a system, the Multi-everything Sonar Simulator (MESS), designed to perform general sonar signal simulations. Fundamental to MESS is a common data structure for describing a particular scenario and interfacing with SST. MESS may be used to simulate any number and variety of sources, receivers, and targets, producing element-level or beam-level data as its output. It may be used to generate a completely synthetic data set, or it may be used, in conjunction with an existing data set, to inject artificial targets into a real environment. The Sonar Simulation Tool (SST) is used as the main processing engine for MESS, which serves as a front end to SST and is implemented in MATLAB in its current form. As such, acoustic modeling is limited to ray-based propagation, which limits applicability to low frequencies scenarios. At present, MESS is still in the initial stages of its development, and there are several possibilities for future work.

Ambient and radiated noise is specified explicitly in terms of a combined narrowband and broadband spectrum. It would be desirable to have these spectra specified parametrically in terms of more physical quantities. Ambient noise may be parameterized by sea state, while radiated flow and engine noise may be parameterized by speed for a particular platform type.

Specification of the ocean environment could also be simplified by using a historical database to map a given reference location and time to a set of appropriate environmental parameters (sound speed profile, bathymetry, sediment type, etc.), such as is done in some modern Tactical Decision Aids.

Finally, target scattering models are a present rather simple, consisting of a few basic geometric shapes (points, spheres, cylinders, and ellipsoids). The current implementation, however, would allow for external target models of greater fidelity to be readily incorporated.

6 Acknowledgments

The authors would like to thank the Applied Research Laboratories of the University of Texas at Austin (ARL:UT) for its support in the initial development of MESS through its 2005 Honor Scholar and High School Apprentice programs. Continuing development has been supported by the Office on Naval Research under Contract No. N00014-00-G-0450-21. The authors gratefully acknowledge Mr. Ali Fakhreddine, a 2005 High School Apprentice, for his work in the early development of MESS. Questions regarding SST were cheerfully answered by Dr. Robert Goddard, and both his assistance and encouragement have been greatly appreciated. DEMUS'04 data was made possible by the Deployable Multistatic Sonar Joint Research Project (JRP), a collaboration between NATO Undersea Research Centre (NURC), MOD (U.K.), and the Office of Naval Research (U.S.).

References

[1] M. B. Porter. Acoustic models and sonar systems. *IEEE J. Oceanic Eng.*, **18**(4):425–437, Oct 1993.

[2] P. B. Prechner and A. N. Bowen. A generalised sonar simulation model. In *IEEE Colloquium on Simulation Techniques Applied to Sonar*, pages 1–4, 19 May 1988.

[3] H. Weinberg. Generic Sonar Model. In *OCEANS*, volume **14**, pages 201–205, Sep 1982.

[4] H. Weinberg. Cass roots. In *OCEANS 2000 MTS/IEEE Conference and Exhibition*, volume **2**, pages 1071–1076, 11–14 Sep 2000.

[5] R. Keenan. An introduction to GRAB eigenrays and cass reverberation and signal excess. In *OCEANS 2000 MTS/IEEE Conference and Exhibition*, volume **2**, pages 1065–1070, 11-14 Sep 2000.

[6] R. P. Goddard. The Sonar Simulation Toolset. In *OCEANS'89. Proceedings*, volume **4**, pages 1217–1222, Sep 1989.

[7] R. P. Goddard. The Sonar Simulation Toolset, Release 4.1: Science, Mathematics, and Algorithms. Technical Report APL-UW 0404, Applied Physics Laboratory, University of Washington, Seattle, Mar 2005. (approved for public release).

[8] B. A. J. Quesson, P. Schippers, J. Groen, J. C. Sabel, and F. P. G. Driessen. Validation of high frequency acoustic target modeling. In *OCEANS 2005 - Europe*, volume **2**, pages 865–871, 20–23 June 2005. An overview of SIMONA may be found at the following URL. `http://www.tno.nl/defensie_en_veiligheid/defensiekennis_en_civiele/onderwaterakoestiek/SIMONA_overview.pdf`.

[9] APL-UW High Frequency Ocean Environmental Acoustic Models Handbook. Technical Report APL-UW 9407, Applied Physics Laboratory, University of Washington, Seattle, 1994. (approved for public release).

[10] R. Urick. *Principles of Underwater Sound*. Peninsula Publishing, Los Altos, 3rd edition, 1983.