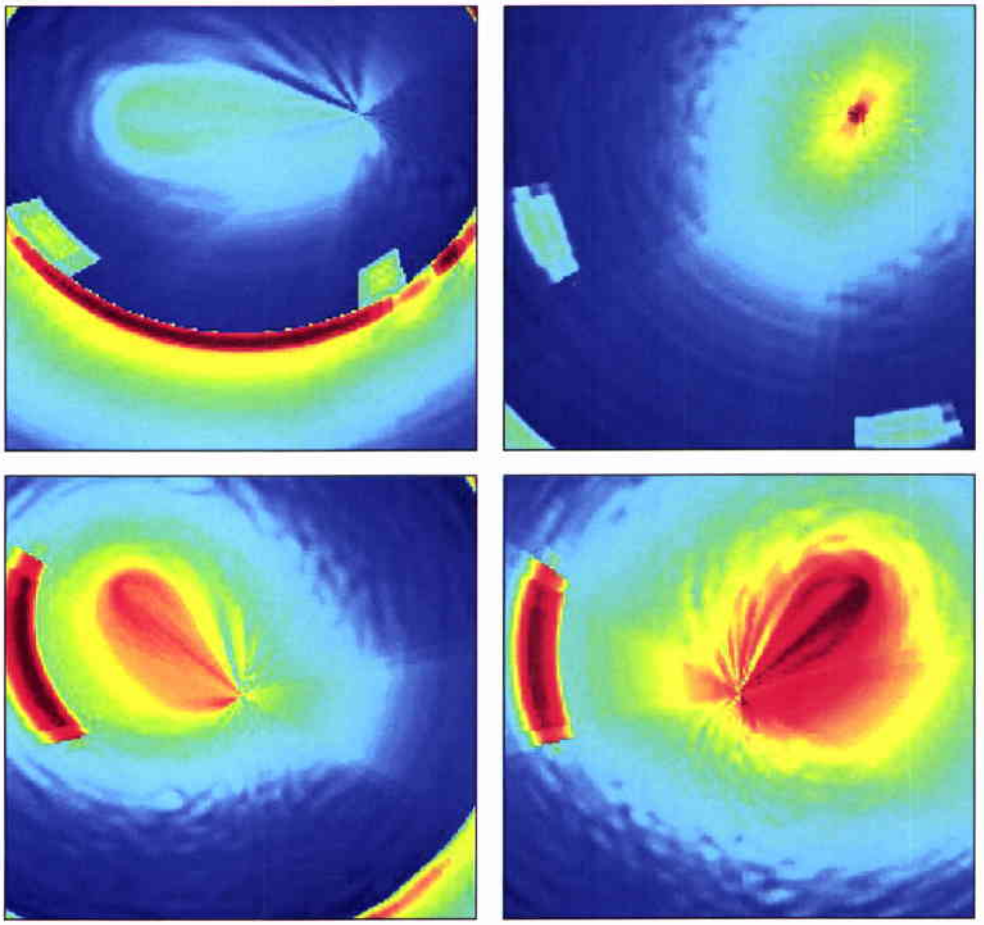
Report no. changed (Mar 2006): SM-396-UU

SACLANT UNDERSEA RESEARCH CENTRE REPORT

SACLANTCEN MEMORANDUM
serial no: SM- 396

SUPREMO:
A multistatic sonar performance model



Chris H. Harrison

November 2002

NORTH ATLANTIC TREATY ORGANIZATION ORGANISATION DU TRAITE DE L'ATLANTIQUE NORD

SUPREMO: A Multistatic Sonar Performance Model

C H Harrison

The content of this document pertains to work performed under Project 04E-1 of the SACLANTCEN Programme of Work. The document has been approved for release by The Director, SACLANTCEN.

Jan L. Spoelstra
Director

intentionally blank page

SUPREMO: A Multistatic Sonar Performance Model

C H Harrison

Executive Summary: Multistatic sonars introduce new problems in operational use such as mutual interference from multiple sources and variable target aspect in a background of range-dependent bistatic reverberation. To help design better trials for multistatic equipment and to investigate new tactics and tactical displays it is essential to model these effects and, typically, existing propagation, reverberation and target models are inadequate for the overall task. A multistatic sonar performance model, SUPREMO, has been developed and built at SACLANTCEN. Its capability includes propagation, target echoes, reverberation and noise, plotted as a function of delay time and bearing, and in equivalent map projection. This document describes the functionality of the model and provides a justification for the algorithms.

A modular propagation section makes it possible to separate the difficulties of propagation modelling (ray/ mode, range-dependence, etc) from those of scatter-area-accounting in delay-time and bearing (computational efficiency, bottom slope, scattering law, etc). Several novel trade-offs of efficiency versus fidelity are discussed.

Special attention is paid to the problems of interference from multiple sources firing in sequence, target aspect dependence from multiple receivers, mixed FM and CW, mismatched source and receivers, multiple displays (one for each bistatic pair), and how to combine the displayed information.

intentionally blank page

SUPREMO: A Multistatic Sonar Performance Model

C H Harrison

Abstract: Multistatic sonars introduce new problems to the modeller such as mutual interference from multiple sources and variable target aspect in a background of range-dependent bistatic reverberation. A multistatic sonar performance model, SUPREMO, has been developed at SACLANTCEN. Its capability includes propagation, target echoes, reverberation and noise, plotted as a function of delay time and bearing, and in equivalent map projection.
A modular propagation section makes it possible to separate the difficulties of propagation modelling (ray/ mode, range-dependence, etc) from those of scatter-area-accounting in delay-time and bearing (computational efficiency, bottom slope, scattering law, etc). Several novel trade-offs of efficiency versus fidelity are discussed.

Special attention is paid to the problems of interference from multiple sources firing in sequence, target aspect dependence from multiple receivers, mixed FM and CW, mismatched source and receivers, multiple displays (one for each bistatic pair), and how to combine the displayed information.

Keywords: Multistatics – reverberation – target echo – propagation – sonar performance – real-time trials planning

Contents

1

Introduction

Although modern targets behave as weak scatterers it is almost inevitable that there will be bright spots or 'glints' in their directional response. One of the benefits of multistatic sonars is that they can take advantage of these glints by planting receivers at many aspects. With multiple source-receiver pairs it is also possible to view the target against a number of reverberation backgrounds, and therefore pick the best signal-to-noise ratio. On the other hand, additional sources bring potential interference and mismatch between source and receiver. Meanwhile the multiple simultaneous geometries create problems of data presentation and recombination in the form of a meaningful tactical display.

SUPREMO is a new model designed from the start to handle the extra "housekeeping" in assessing the performance of such a sonar [1-5]. This report is essentially a functional specification; it briefly describes SUPREMO's design and performance. It also includes discussion of some features such as ambient noise and Doppler which are not yet implemented. The main structure is in MATLAB which runs the addition-over-scatterers section as a FORTRAN executable. The propagation calculations can be run by an external model, if desired. The general philosophy is to allow the user to trade CPU time for approximation in a controlled way (incidentally demonstrating the validity of the approximations) rather than compromising the solution by embedding approximations at the outset. As a result, realistic solutions are available in seconds rather than hours.

Relevant features are summarised as follows. Platforms are assumed to carry an arbitrary numbers of sources and receivers. Each source has a finite bandwidth, a beam pattern, a pulse envelope, and an emission time in an overall repeating sequence. Each receiver has an independent bandwidth and a linear array or beam pattern. The model is designed to be fully range/azimuth-dependent although currently multipath propagation is handled by range-independent codes GAMARAY [6], PROSIM [7] or an internal image calculation. Scattering strength is a function of position and local angles, and the effect of local bottom slope on scattering is included. Calculations include targets and false targets, represented as two-dimensional functions of angle, diffuse reverberation (bottom, surface, volume), geoclutter (ridges, etc), ambient noise and self noise of the known friendly ships.

At a high level the program structure is very simple. First the propagation from each source and from each receiver location is tabulated. Then for each possible bistatic pair a target calculation and a reverberation calculation (bottom, surface and volume) result in total response versus absolute time (or delay time relative to any particular source). For each receiver there will eventually be an ambient noise calculation based on wind, known local ships and a shipping background. Finally one can generate displays in all the standard forms, and some are shown later. Naturally the display for each source/receiver

combination may include interfering contributions from the other sources; this is precisely the novelty of multistatic assessment.

2

Propagation and reverberation

2.1 Propagation

Whether the external propagation model is range dependent or independent it must supply a set of intensities I_n, elevation angles θ_n, and delay times t_n for each range (and bearing). The set can consist of strict eigenrays or just sampled intensity. Conversion work is required for wave models. These three parameters are tabulated non-linearly against range to allow for more detailed representation at close range. In anticipation of the scattering calculation, where scattering strength is a separable function of angle-in and angle-out, a table of 'modified intensity', eg $I_n \times \sin(\theta_n)$ is also built.

A crucial fidelity issue is the inclusion, for each scatterer, of all combinations of eigenrays-in and eigenrays-out. To retain fidelity but avoid excessive CPU times it is possible to 're-sample' the eigenrays whilst keeping the total energy unchanged. Regarding them as a continuous function of time and angle they can be sampled progressively more coarsely, at the user's discretion. In this way one can see whether the extra fidelity was really justified.

2.2 Reverberation

Here we concentrate on bottom reverberation although surface and volume modules function in a similar fashion. There are three calculation stages, as follows:

- *Stage 1: Pre-process Scatterers*

In principle, numerical scatterers can be regarded as points or contiguous rectangular or triangular facets. Only the rectangles are forced to be on a grid. The model takes its scatterers from a list (in any order) giving, for each scatterer, x,y location and product of facet area times local (interpolated) scattering strength. However the user always defines scattering strengths at the intersection points of a rectangular grid. One user-option is to define scatterer locations as the same grid points. Although area accounting is correct, there is a reverberation sampling problem; at long range there is an unnecessarily large number of weak scatterer contributions (computational overkill), and at short range an unacceptably small number of strong scatterers (spiky artifacts). The "pre-process" (for details, see Section 5) is a method of producing a more uniform distribution of calculation points in bearing-time space.

• *Stage 2: Calculate scatterer contributions; Allocate to time-bearing bins*

Given the revised set of 'eigenrays' and the scatterer list, the model gets the intensity contribution from every combination of scatterer, eigenray-in and eigenray-out as $I_n \times I_m \times S_i(\theta_n , \theta_m , \beta) \times A_i$, where the (linear) separable scattering strength of the *i*th facet, of area A_i , is $S_i(\theta_n,\theta_m,\beta) = S_{IN}(\theta_n) \times S_{OUT}(\theta_m) \times S_{BIS}(\beta) \times F_i$, and F_i represents geographic variation. The arrival bearing and elevation angle are converted to equivalent beam angle α at the receiving array. Since the round-trip travel time is known the intensity contribution can be added to the appropriate time-bearing bin. The result is a reverberation intensity array versus time and bearing.

• *Stage 3: Post-process in time-bearing space*

If we had used a rectangular pulse of the same length as the *time* bin separation and rectangular beams of the same width as the *angle* bin separation the allocated scattering area would be exactly correct. (Note that for a linear array we need bin width proportional to $\sin(\alpha)$). For other cases we need to compensate in both time and angle; here we discuss time only. Suppose we use very fine bins, so that we frequently find empty ones. With appropriate normalisation we can find the response to any pulse (envelope) shape by convolution. We do this in the frequency domain as a post-process by storing the FT of the desired pulse, multiplying by the FFT of the reverberation and then IFFTing back to time. For each source/receiver pair this extra FFT and IFFT only needs to be done once after all reverberation and target contributions have been inserted into the master array. It therefore takes a negligible time since it is always much faster than any of the acoustic calculations. Note that high time resolution is only required at the leading edge of the reverberation and for rapidly varying scatter features (ie "geo-clutter"). A future variant of this process will allow the combination of low resolution "diffuse" reverberation and high resolution "geo-clutter" to maximise efficiency.

3

Time domain / frequency domain theory

3.1 Definitions

Intuitively it seems plausible that convolving the 'pulse' with the 'arrivals' gives us the reverberation response to an FM pulse. The objective of this section is to derive the acoustic solution rigorously.

The true solution for reverberation is to start with FM pulse (oscillatory but not complex), to propagate through the sea (ie convolve with sea's impulse response, including scatterers etc), possibly to filter at the receiver (ie convolve with another function), to correlate with a (potentially different) FM pulse in the receiver, to rectify and smooth (get envelope, eg by deriving the imaginary part from the real part using Hilbert transform squaring and adding), and finally to display some function of the envelope. This function of time can be echo, reverberation etc on an absolute scale but normalised according to some convention.

In the time domain we have

$$a(t) = p_S(t) \otimes q(t) \otimes f(t) \otimes p_R{}^*(-t) \tag{3.1}$$

The "steering wheel" symbol means convolve. Note that you turn alternate functions around in time for all except the correlation. p_S, p_R, are FM pulse waveforms, q is the impulse response of the sea, f is the impulse response of a filter (assumed) in the receiver. Strictly these are all real quantities because they are real input/output volts. Phase is $\cos(\omega t+\phi)$; the amplitude is still real. Note that the impulse response q of the sea *really* means impulse response in this context! One cannot talk of pulse arrivals at one frequency with this function.

The Convolution theorem for complex f,g [8] is

$$c(\tau) = \int f(t)g(\tau - t)\,dt$$
$$C(\omega) = F(\omega)G(\omega) \tag{3.2}$$
$$\textit{where} \quad J(\omega) = \int j(t)\exp(i\omega t)\,dt; \quad j(t) = \frac{1}{2\pi}\int J(\omega)\exp(-i\omega t)\,d\omega$$

The Wiener-Khintchine theorem for cross-correlation function is similar (as can be shown by putting $J^*(\omega) = G(\omega)$ in the above).

$$k(\tau) = \int f(\tau + t) g^*(t)\, dt$$
$$K(\omega) = F(\omega) G^*(\omega) \tag{3.3}$$

As long as the system is linear (it won't be for Doppler, Annex B) the terms in Eq (3.1) and Eq (3.2) can be rearranged, eg

$$a(t) = (p_S(t) \otimes p_R^*(-t) \otimes f(t)) \otimes q(t) \tag{3.4}$$

So we might as well do the pulse correlation straight off (in either the time or frequency domain). The result is a new high resolution pulse p

$$p(t) = p_S(t) \otimes p_R^*(-t) \otimes f(t) \tag{3.5}$$

and

$$a(t) = p(t) \otimes q(t) \tag{3.6}$$

In the frequency domain

$$A = PQ \tag{3.7}$$

Expanding Eq (3.6) we have

$$\begin{aligned} a(t)a^*(t) &= \int p(t') q(t-t')\, dt' \int p^*(t'') q^*(t-t'')\, dt'' \\ &= \int\int p(t')\, p^*(t'')\, q(t-t')\, q^*(t-t'')\, dt'\, dt'' \end{aligned} \tag{3.8}$$

We now invoke *incoherence*, meaning that the environmental impulse response q(t) can be represented as $q_{env}(t) \times \phi(t)$, where $q_{env}(t)$ is a relatively slowly varying envelope (still an amplitude) and $\phi(t)$ is a chaotic[1] part with peak value unity. We find that $q(t)q^*(t') = q_{env}(t) q_{env}^*(t')\, \phi(t)\phi^*(t')$. Since $\phi(t)$ is chaotic the mean value of $\phi(t)\phi^*(t')$, ie $\langle \phi(t)\phi^*(t') \rangle$ is $|\phi(t)|^2 \times \rho_\phi(t - t') = |\phi(t)|^2 \times \delta(t-t')$. Therefore Eq (3.8) becomes

$$\begin{aligned} \langle a(t)a^*(t) \rangle &= \int\int p(t')\, p^*(t'')\, |q_{env}(t-t')|^2\, \delta(t'-t'')\, dt'\, dt'' \\ &= \int |p(t')|^2\, |q_{env}(t-t')|^2\, dt' \\ &= \int i_p(t')\, i_q(t-t')\, dt' \\ &= i_p \otimes i_q \end{aligned} \tag{3.9}$$

[1] Here by "chaotic" we mean "periodic with slight perturbations so that after many cycles the phase is pretty uncertain". Thus the function itself is not random.

where $i_p(t)$ and $i_q(t)$ are time domain volt-squared quantities (time domain/frequency domain distinction is important if frequency components are correlated, see Annex A). Equation (3.9) is the basis of convolving the post-correlator pulse envelope with the time domain intensity arrivals. So the above is a rigorous proof that we are doing the correct thing.

Since the approach deals with a finite band we also have to cope with variation of propagation loss and beam pattern across the band. These are explained in Annexes C and D.

4

Reverberation: exact intensity

4.1 Areas and contributions in bins

If we map a square x,y grid into time-bearing space, having assumed separated source and receiver, it looks like this.

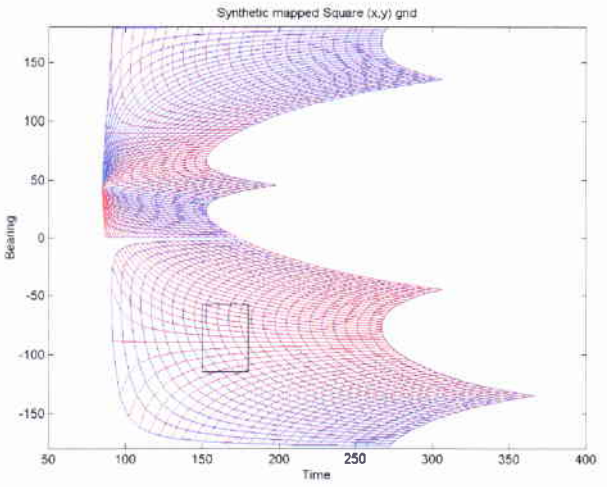


Figure 4.1 *An x,y grid mapped into time-bearing space (x-lines are red; y-lines are blue)*

Now suppose each $\delta x\ \delta y$ square is a scattering facet. The power attenuation from source to scatterer is I_s (one-way transmission loss = $10 \log_{10} (I_s)$); similarly the power attenuation from receiver to scatterer (using reciprocity) is I_R. Therefore the power contribution to reverberation is

$$R = I_s I_R S \ \delta x\ \delta y \qquad (4.1)$$

where S is the linear power scattering strength (a function of whatever you want). Because SUPREMO knows the delay time and the bearing for each scatterer contribution (no matter how convoluted its path is) it allocates this power to a dot in the middle of the appropriate (red/blue) representation of an x,y square. However it accumulates all the contributions in rectangular (black) $\delta t\ \delta\phi$ bins, ie on a rectangular t,ϕ grid. Thus in the picture shown there are several $\delta x\ \delta y$ areas that contribute to the same bin. If we had used

a "top-hat [UK]" / "boxcar[US]" pulse of length exactly δt and a "top-hat" beam of width exactly $\delta\phi$ we should get a power that corresponds exactly to $\delta x\ \delta y$ times the number of red/blue 'squares' in the black square. In other words the mapping combined with putting contributions into bins takes care of the Jacobian for the mapping by itself; if there are many red/blue squares in the black square we automatically get more power. If we go to longer range the same black square covers more red/blue squares.

The following two pictures (Figs 4.2 1nd 4.3) show that there is no problem if we use a different mapping. In this case there is a horizontal array so we measure angles relative to the array axis and then we take the cosine (or sine of the angle from broadside). This introduces left-right ambiguity, and so two parts of the x,y plane map to the same t, $\sin(\phi)$.

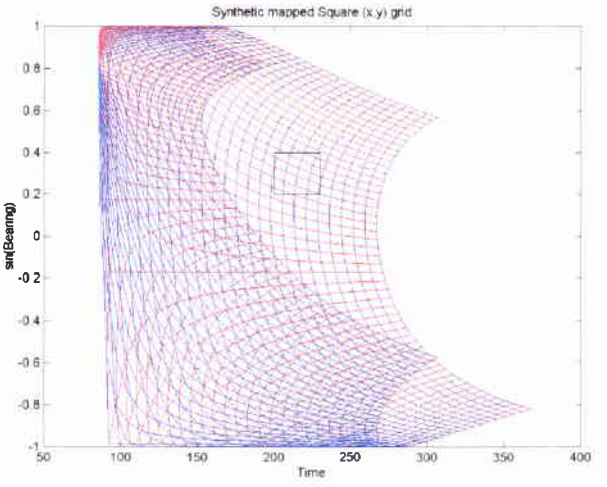


Figure 4.2 *An x,y grid mapped into time-sine-of-bearing space – example 1.*

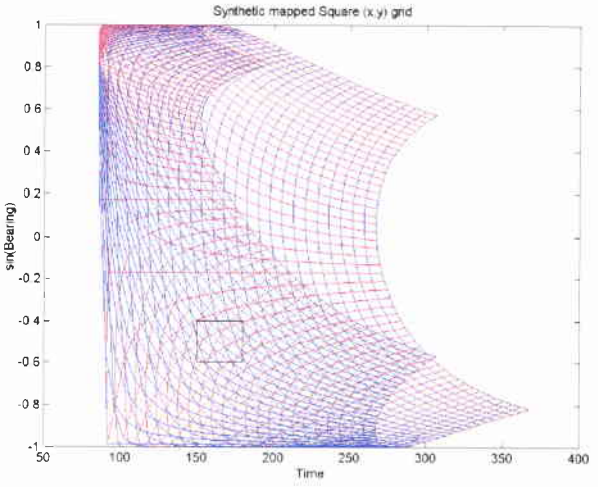


Figure 4.3 *An x,y grid mapped into time-sine-of-bearing space – example 2*

What happens when the $\delta t\ \delta\phi$ bins are small or smaller than the red/blue squares is shown in Fig 4.4.

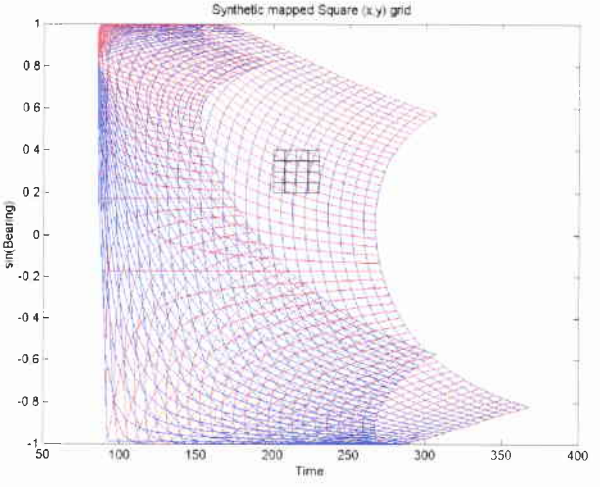


Figure 4.4 *An x,y grid mapped into time-sine-of-bearing space – small time-bearing bins*

This is just the histogram-bin-size problem; some bins will remain empty while others get all the contributions. On average everything evens out, but, better still, the convolution post-process described below completely gets rid of these problems.

4.2 Scattering Laws

A number of scattering laws are provided including: angle-independent, Lambert's Law, Lommel-Seeliger, tilted facets each obeying Lambert's Law. Because the angle-dependence exists in the Matlab part of the code it is easy to construct any desired behaviour.

4.2.1 Angle-independent scatterers

Scattering is assumed to be independent of the incoming and outgoing angle.

4.2.2 Lambert on a flat plane

We assume that Lambert's Law is separable in incoming and outgoing angle so that the scattering strength is given by

$$S = \mu \sin\theta_{in} \sin\theta_{out}$$

and the coefficient μ is dependent on position.

4.2.3 Lommel-Seeliger on a flat plane

The Lommel-Seeliger Law is not separable and S is

$$S = \mu \frac{\sin\theta_{in} \sin\theta_{out}}{(\sin\theta_{in} + \sin\theta_{out})}$$

4.2.4 Tilted facets each obeying Lambert's Law

Here the incoming and outgoing angles are measured relative to the tilted plane of the facet. Surprisingly, since bottom slopes are always small the extra trigonometry reduces to an almost trivial change.

The normal to the facet is at θ_n, ϕ_n and these can be written in terms of the x and y derivatives of the bathymetry H(x,y). The new incident and scattered angles are given by

$$\sin\theta'_{in} = \sin\theta_{in} \sin\theta_n + \cos\theta_{in} \cos\theta_n \cos(\varphi_{in} - \varphi_n)$$
$$\sin\theta'_{out} = \sin\theta_{out} \sin\theta_n + \cos\theta_{out} \cos\theta_n \cos(\varphi_{out} - \varphi_n)$$

Assuming that $\theta_n \approx \pi/2$, hence $\sin\theta_n \approx 1$ and that $\theta_{in} \approx \theta_{out} \approx 0$, *i.e.* $\cos\theta_{in} \approx \cos\theta_{out} \approx 1$; the above relationships become (only the first one is developed here):

$$\begin{aligned}
\sin\theta'_{in} &= \sin\theta_{in} + \cos\theta_n\left(\cos\varphi_{in}\cos\varphi_n + \sin\varphi_{in}\sin\varphi_n\right) \\
&= \sin\theta_{in} + \cos\varphi_{in}\left(\cos\theta_n\cos\varphi_n\right) + \sin\varphi_{in}\left(\cos\theta_n\sin\varphi_n\right) \\
&= \sin\theta_{in} + \cos\varphi_{in}\left(-\frac{dH}{dy}\right) + \sin\varphi_{in}\left(-\frac{dH}{dx}\right)
\end{aligned}$$

The resulting scattering law is hence:

$$S = \mu \times \left(\sin\theta_{in} - \cos\varphi_{in}\frac{dH}{dy} + \sin\varphi_{in}\frac{dH}{dx}\right) \times \left(\sin\theta_{out} - \cos\varphi_{out}\frac{dH}{dy} + \sin\varphi_{out}\frac{dH}{dx}\right)$$

As a check, it is easy to see that the corrected angle increases as the ray hits an upward slope.

4.2.5 Comparison between scattering laws

Figure 4.5 shows a comparison between reverberation levels as a function of range with three different scattering laws applied to the same environment. As expected, the reverberation level for a given range decreases as we pass from angle-independent through Lommel-Seeliger to Lambert's Law.

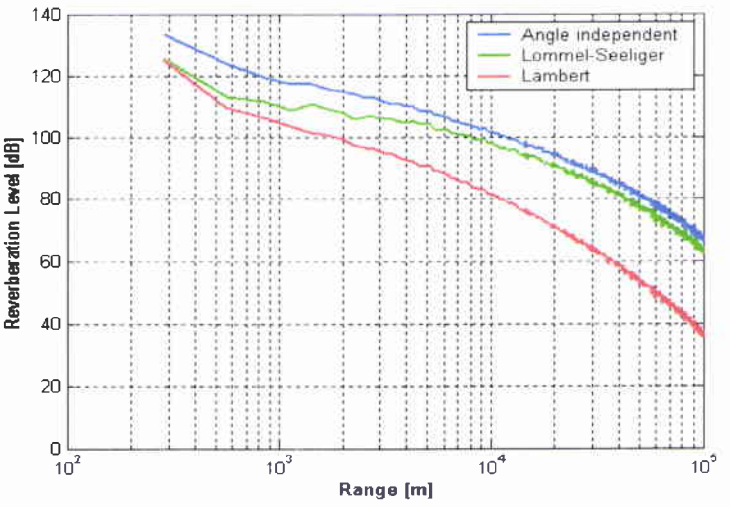


Figure 4.5 *Comparison between different scattering laws in a range-independent environment*

5 Delaunay triangles

If we were to use a regular x,y grid of scattering points there would be a sampling problem; at long range there would be an unnecessarily large number of weak scatterer contributions (computational overkill), and at short range an unacceptibly small number of strong scatterers (spiky artifacts). What we really want is a uniform density of scatter contributions in time-bearing space. One way (of several) of doing this is to set up a uniform random distribution of points in time-bearing space then map them back to the seabed using a crude horizontal plane transformation. Taking these dots as the starting point we join them with Delaunay triangles [12] to form a surface of triangular facets (Fig 5.1, note concentration near receiver (10,10)). We then calculate x,y for each facet centre, and the product of facet area and interpolated scattering strength.

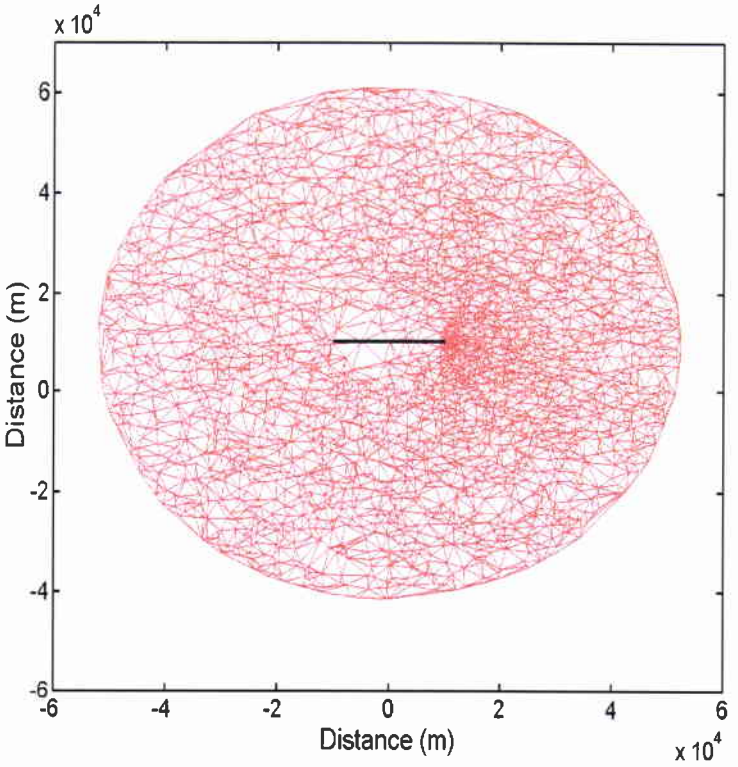


Figure 5.1 *Delaunay triangular scatterers in x,y space. Black line joins source and receiver. Note concentration of triangles at receiver on right.*

Although a new calculation is required for each source/receiver combination the entire operation takes only a fraction of a second for several thousand scatterers, and therefore CPU time is quite negligible. The effect on time-bearing is, at worst, a superimposed weak random ripple spanning all ranges. Of course this is merely a numerical device and any required spatial detail must be resolved by the scattering strength grid.

5.1 Mapping of (ϕ,t) to (x,y)

Note that this mapping is only a rough one in the horizontal plane to define scattering points with a suitable distribution in x,y. It is distinct from the transformation done in the geographic plot where the contents of the bearing-time bins (ϕ,t) are replotted at a point in (x,y). Note also that the travel times t are calculated along the multipath rays by the propagation module.

Given a uniformly distributed delay time t and bearing ϕ (defining an apex of a triangle), the distance L between receiver and source, and the bearing from receiver to the source ϕ_s we calculate the range r to the point by the cosine rule as

$$r = \frac{(ct)^2 - L^2}{2(ct - L\cos(\phi - \phi_s))} \tag{5.1}$$

It is then easy to calculate x,y.

6
Post-processing

Post-processing is carried out after all scattering calculations are complete. The same operation is applied in both the time and the bearing dimensions.

6.1 Time post-processing

It is clear from the arguments in Section 4.1 that if we really use a longer pulse than the bin size we get more power, in the ratio

$$\int |p_{env}(t)|^2 dt \Big/ \left(|p_{env}(t_{\max})|^2 \, \delta t \right) \tag{6.1}$$

where p_{env} is the amplitude envelope of the pulse, t_{max} is the point where the maximum power is, and δt is the bin size. Notice that pulse amplitude (or power) is normalised out leaving the correct dimensions as in Eq (4.1), earlier. Notice also that there is no restriction on the bin size; we can make it as small as we like, so that the arrivals, though a poor histogram, make a good representation of an impulse response.

To see the effect of a power pulse envelope on the power impulse response of the environment we convolve the pulse shape with the impulse response.

$$R(t_s) = \int P(t_s - t) I(t) \, dt \Big/ \left(P(t_{\max}) \, \delta t \right) \tag{6.2}$$

where $P(t) \equiv |p_{env}(t)|^2$. In DFT terms, remembering that the bin size δt is the sample interval, we have

$$R_n = \sum_m P_{n-m} I_m \Big/ P(t_{\max}) \tag{6.3}$$

So really we require a unit amplitude pulse shape to conform with Eq (4.1). The convolution is easily done in the frequency domain by taking FFTs of the two time domain quantities, multiplying, and IFFTing back to time. In this way δt has no significance other than determining the size of the FFT; for instance, the size of δt has no implication on the required amount of acoustic calculation, this is determined purely by the number of scatterers and eigenrays.

6.2 Bearing post-processing

Rays arrive from a single scatterer at the receiver location along many paths in the vertical plane, i.e. each contribution has a well-defined elevation and azimuth θ,ϕ. What constitutes a beam or a bearing depends on the array type and orientation. A planar-type array can be steered from 0 to 360 degrees *around* its axis. A line array can only be steered from 0 to 180 degrees *from* the axis. Any lack of symmetry results in an ability (in principle) to steer fully in 3D. Whatever happens, it is straightforward to calculate a set of steered beam patterns for the given array. We then wish to display echoes and reverberation against one (or two, if available) steer angles and delay time. All we do is convert the well-defined elevation and azimuth θ,ϕ into the appropriate steer angles α,β. All ambiguity is automatically included by the (single-valued or double-valued) conversion from θ,ϕ to α,β.

6.3 Angle post-processing

Post-processing in angle can be done in *exactly* the same way as in time although there are subtle differences between planar arrays and linear arrays.

In the case of a planar array we have essentially the same beam shape $B(\phi)$ anywhere in the plane of the array, but the centre can point in different directions. Therefore the response to the contents of the bins $I(\phi)$ is exactly a convolution.

$$R(\phi_s) = \int B(\phi_s - \phi) I(\phi)\, d\phi \tag{6.4}$$

Again this convolution is executed in the frequency domain, and since angles repeat after 360 degrees the FFT wrap-around effects are correct and legitimate.

In the case of a line array it is easy to show that the beam pattern (being a plot versus angle α for a steer angle α_s, with α measured from broadside) is always a function $B(\sin(\alpha)-\sin(\alpha_s))$. If the angle around the array axis is β then

$$\begin{aligned} R(\alpha_s) &= \int B(\sin(\alpha_s) - \sin(\alpha)) I(\alpha,\beta)\ \cos(\alpha)\, d\alpha\, d\beta \\ &= \int B(X_s - X) \{ \int I(X,\beta)\, d\beta \}\ dX \end{aligned} \tag{6.5}$$

This is again an exact convolution which we can execute in the frequency domain. However we need to be careful of wrap-around effects. In reality they don't wrap from one endfire to the opposite one. Therefore we need to include a section of one half-beamwidth's worth of zeros at each end of I(X) before we carry out the FFTs. This gives the correct behaviour as the endfire beam turns into oversteer.

The calibration question has an equivalent answer to that in the time dimension, namely that we ensure that the angle bins (in ϕ or $\sin(\phi)$) are always much smaller than the true

beam width so that the x,y area (red/blue squares in Figs 4.1-4.4) is determined by beam width but there is no change to the amplitude. Thus the convolutions are normalised as

$$R(\phi_s) = \int B(\phi_s - \phi) I(\phi)\, d\phi \Big/ \left(B(\phi_{\max})\, \delta\phi\right) \tag{6.6}$$

where ϕ_{max} is the centre beam direcction. In DFT terms, remembering that the bin size $\delta\phi$ is the FFT sample interval, we have

$$R_n = \sum_m B_{n-m} I_m \Big/ B(\phi_{\max}) \tag{6.7}$$

So really we require a unit amplitude beam shape.

6.4 Joint angle-time post-processing

A two-dimensional version of the function `postproc` does the FFT of I(ϕ,t) in two dimensions at once using the Matlab function FFT2. The correct normalisation for the 2D FFT is 1 / [P(t_{max}) B(ϕ_{max})].

6.5 Mapping of (ϕ,t) to (x,y) in geographic plots

Mathematically this is similar to the mapping used in Section 5. However there is a difference between the mapping for a horizontal planar array and that for a linear array. It is precisely this that is responsible for the left-right ambiguity in the linear case, and the relative orientation of the array axis and the source affects the amount of asymmetry in the beam response.

It is easieast to take each pixel (x,y) in the picture, convert it to ϕ,t using simple geometry, then look up intensity (which has already been calculated as a function of ϕ,t). In the case of a planar array the intensity was indeed mapped directly against bearing ϕ. For a linear array (any orientation) the original scattering code determined an intensity and a vertical and horizontal arrival angle for each scatterer and each incident ray. At that stage it was possible to calculate the 3D arrival angle relative to the array axis. The intensity was therefore allocated to bins in (array-axial-angle,time) space. The geographic plot, however, knows nothing of the vertical angle and we therefore treat the given array-axial-angle as if it were a horizontal angle from the array axis to look up the intensity. This action alone produces the well-known array steering ambiguities.

7

Further calibration issues

7.1 Noise received through the correlator

We assume the incident noise power spectral density is given as $N(\omega,\theta,\phi)\cos\theta\, d\theta\, d\phi$ and that the processing characteristics are still as in Eq (3.5), but excluding the source part, ie only $p_R{}^*(-t) \otimes f(t)$. In the frequency domain we get

$$P(\omega) = \int N(\omega,\theta,\phi)\left|P_R(\omega)\right|^2 \left|F(\omega)\right|^2 d\omega \cos\theta\, d\theta\, d\phi \qquad (7.1)$$

This is equivalent to multiplying by the bandwidth as one does in the sonar equation.

7.2 Check-sum / normalisation

In Eq (3.9), where the convolution originates, the effective pulse p (which results from cross correlating p_S,p_R,f in Eq (3.5)) has a large amplitude which is the whole point of correlation processing. In Eq (6.3) we started with a unit amplitude pulse, in effect, a simple sonar without correlation processing but with the same bandwidth. Clearly there is a conversion from one to the other and finally an arbitrary normalisation. Here we derive the conversion and justify the normalisation by considering the computed echo, reverberation, and noise levels for the two sonars.

7.2.1 Correlation sonar

If the correlation sonar has $p_S(t) = p_R(t) = \exp(-it(\omega_0 + tB/T))$, where B is bandwidth and T is pulse length, then the effective pulse $p(t) = \int p_S{}^*(t')\, p_R(t+t')\, dt'$ has amplitude T at its maximum. Thus the effective pulse is height T and width $t_p = 1/B$. This differs from Eq (6.3) in that the pulse amplitude is multiplied by T. Therefore all echoes and reverberation powers are multiplied by T^2. Noise only sees the effect of p_R. In the frequency domain it sees the band pass $\frac{1}{2\pi}\int\left|P_R(\omega)\right|^2 d\omega$ which by Parseval's theorem can be written as $\int\left|p_R(t)\right|^2 dt$. From the above definition of p_R this is just T.

7.2.2 Simple sonar: unit amplitude pulse

We now mimic a simple sonar using the correlation sonar. To do the following rough calculation we assume the effective pulse to be gaussian shape. We want a unit amplitude source pulse with the same length and bandwidth as the true correlation sonar's effective pulse. We also need a receiver with unit gain (in the frequency domain), ie

$p_S'(t) = \exp(-t^2 / t_p{}^2)\exp(-i\omega_o t)$ (correct unit amplitude and pulse length t_p)

$P_R'(\omega) = \exp(-(\omega - \omega_o)^2 t_p{}^2 / 4)$ (receiver has no gain; band pass matches source band)

where dashes indicate simple sonar parameters.

So all echoes and reverberation powers are multiplied by unity since the pulse is unit amplitude.

Taking respective FTs we have

$$P_S'(\omega) = \sqrt{\pi}\, t_p \exp(-(\omega - \omega_o)^2 t_p{}^2 / 4)$$

$$p_R'(t) = 1/(\sqrt{\pi}\, t_p)\exp(-t^2 / t_p{}^2)\exp(-i\omega_o t)$$

Again noise only sees the effect of p_R. In the frequency domain it sees the band pass $\frac{1}{2\pi}\int |P_R'(\omega)|^2 d\omega$ which can be evaluated as $1/(\sqrt{\pi}\, t_p)$

7.2.3 Summary

The proportionality factors are summarised as follows:

	Echoes	Reverberation	Noise
Correln. Sonar	T^2	T^2	T
Simple Sonar	1	1	$1/(\sqrt{\pi}t_p)$

Then we normalise such that the noise power in the band is unchanged, and we retain the dimensions of all quantities. In other words, to normalise the correlation sonar using exact quantities derived from the actual pulses we *multiply* by

$$\int |P_R'(\omega)|^2 d\omega \Big/ \int |P_R(\omega)|^2 d\omega$$

The numerator (for the simple sonar) can be written conveniently in terms of the correlation sonar as

$$\int |P_R'(\omega)|^2 d\omega = \int |P_R(\omega)|^2 d\omega \Big/ \mathrm{MAX}(|P_R(\omega)|^2)$$

So the whole normalisation reduces to simply

$$Norm = 1/\mathrm{MAX}(|P_R(\omega)|^2) \tag{7.2}$$

Although not obvious, this factor is roughly B/T. The table becomes (after multiplying the top row by $1/(\sqrt{\pi t_p}T)$, ie B/T):

	Echoes	Reverberation	Noise
Correln. Sonar	$T/(\sqrt{\pi t_p}) = BT$	$T/(\sqrt{\pi t_p}) = BT$	$1/(\sqrt{\pi t_p})$
Simple Sonar	1	1	$1/(\sqrt{\pi t_p})$

The signal-to-noise gain of a correlation sonar is BT above a simple sonar as expected.

The outcome of all this is that two options are available to the user. One option is the normalised correlator output (that corresponds to the fictitious high source level of the effective pulse); the other is the un-normalised correlator output that corresponds to the real acoustic power in the water (in the common source/receiver band). This might be useful for instance, for calculating source emission levels in the context of inadvertent receivers such as whales and divers.

To make the correlation sonar code produce the simple sonar result we normalise so that the effective pulse is unit amplitude, ie

$$p_{Eff}(t) = \int p_S(t') p_R(t + t') dt'$$

$$Norm = 1/\mathrm{MAX}(|p_{Eff}(t)|^2) \tag{7.3}$$

From the programmer's point of view the effective pulse is defined in both cases by the correlation between the real source and receiver pulse (eq 5, done in the frequency domain). We then apply either Eq (7.2) or (7.3) to normalise.

8 Displays

Here we concentrate on the information available to be displayed and the novelties of multistatic sonar rather than exact display formats. In this example the scenario consists of three numbered ships shown as red stars on the map (Fig 8.1). Initially we consider only one source on ship 1 but the two receivers on ships 2 and 3. The green square is a target and there is otherwise a uniform scattering background. The most straightforward way of viewing the results is as a Cartesian plot of time vs bearing (actually steer angle at the receiver) as shown in Fig 8.2. Since we know the intensity and travel time for each bearing we can also plot the same data on maps (geo-plots) as in Fig 8.3. The black arrow shows the orientation of the array, and one can see a strong cigar-shaped return pointing towards the source. Clear ambiguous beam returns from the target can be seen in the ship 2 response but not in the ship 3 response because the latter are near to endfire. This ambiguity and broadening of the endfire beam occur quite naturally without any modification of the model algorithms.

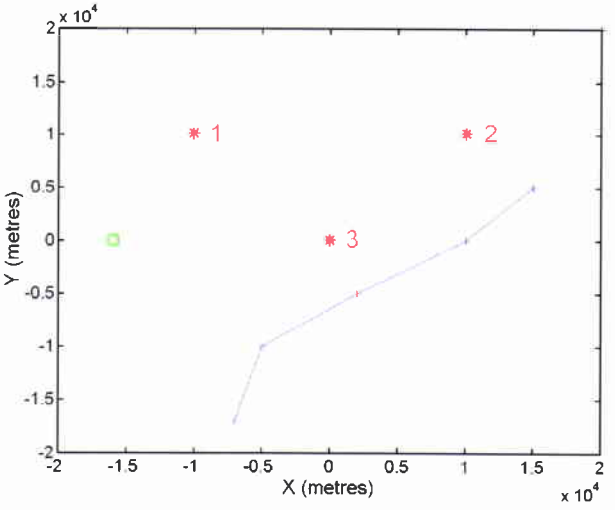


Figure 8.1 *Map showing three ships (numbered red stars), a target (green square) and a possible target track (blue). Ships 1 and 2 each have a source; ships 2 and 3 each have a receiver.*

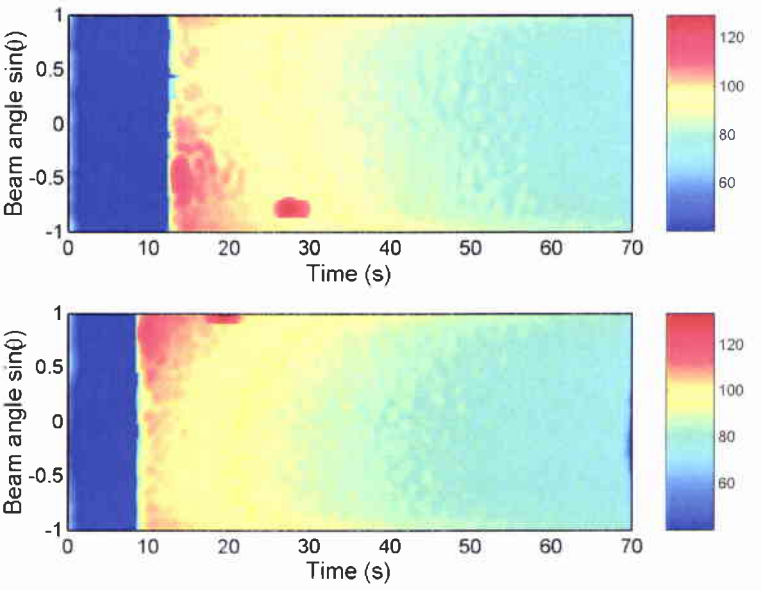


Figure 8.2 *Cartesian plot of arrival intensity vs delay time and sine of the steer angle. Upper plot is receiver 2 viewing source 1; lower plot is receiver 3 viewing source 1. Note longer delay of first reverberation arrival on receiver 2, and target received at endfire on receiver 3.*

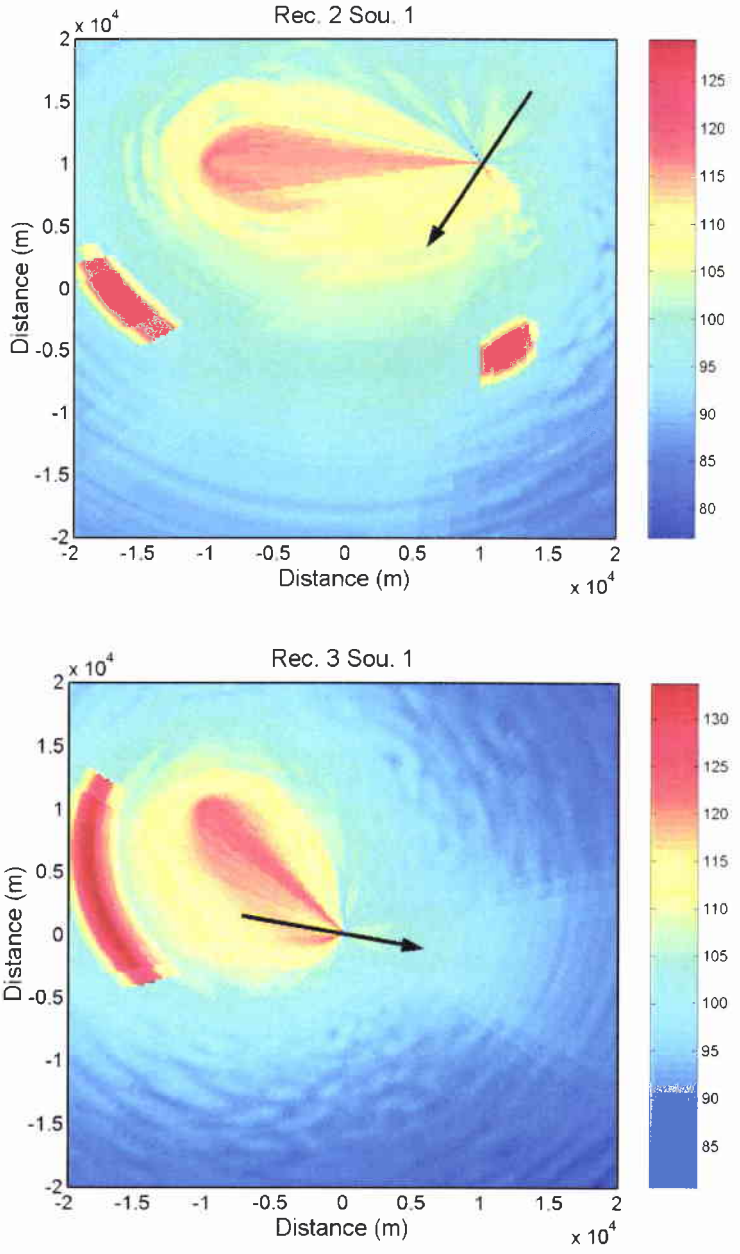


Figure 8.3 *Geo-plots for source 1 viewed by receivers 2 and 3. The black arrow shows the orientation of the array, and one can see a strong cigar shaped return ("direct blast") pointing towards the source. Clear ambiguous beam returns from the target can be seen in the ship 2 response but not in the ship 3 response because the latter are near to endfire.*

Next we take a more complicated scenario where we include the two sources and receivers from Fig 8.1. Now we assume that the two sources ping in an endless sequence repeating every 70 seconds with source 2 occurring 30 seconds after source 1 (and

therefore source 1 occurring 40 seconds after source 2). The Cartesian views from receivers 2 and 3 are shown in Fig 8.4. For each receiver we see the direct blast from the two sources and their respective displaced target echoes. Notice that, to receiver 2, source 2 appears much stronger than source 1 because it is effectively monostatic.

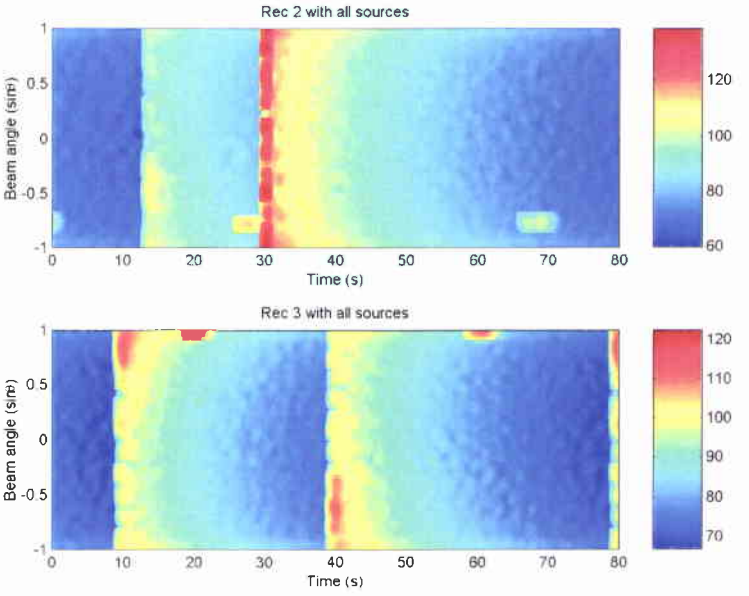


Figure 8.4 *Cartesian plot of arrival intensity vs delay time and sine of the steer angle. Upper plot is receiver 2 viewing source 1 and 2; lower plot is receiver 3 viewing source 1 and 2.*

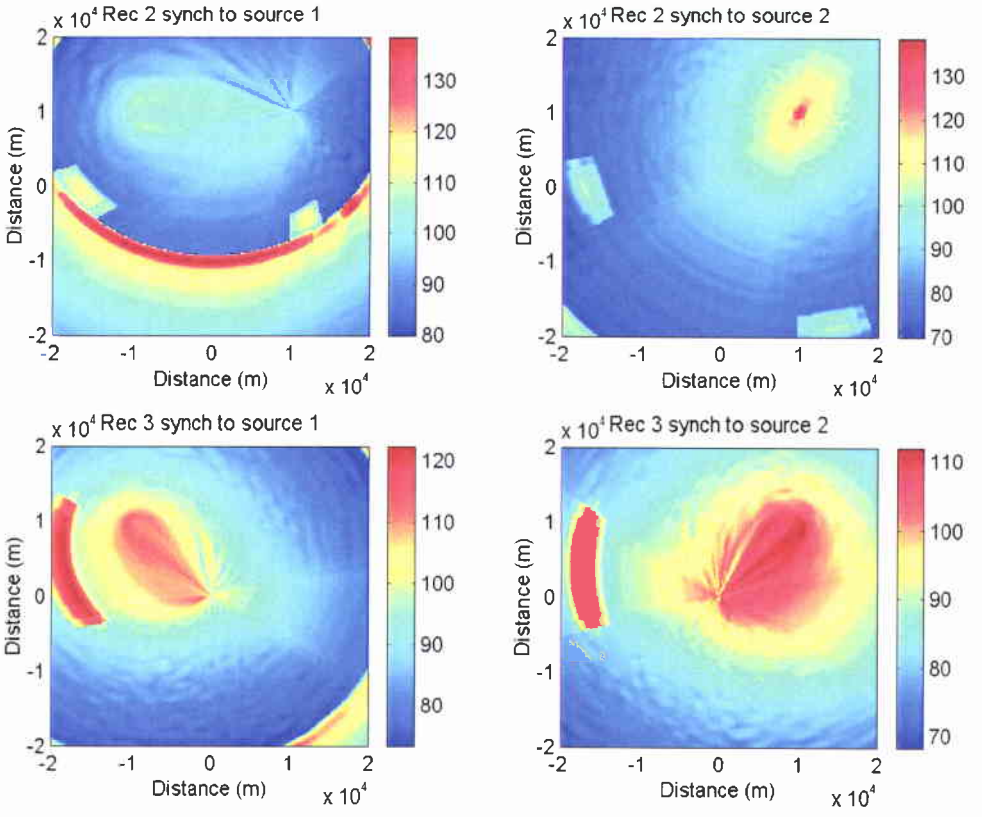


Figure 8.5 *Geo-plots for all source-receiver pairs. Again one can see a strong cigar-shaped return pointing towards the source except in the top right picture which is monostatic. Again there are ambiguous beam returns for receiver 2. Note also the interference in the form of elliptical ghost returns from the direct blast of the other source.*

In order to convert these two Cartesian plots to geo-plots we have to decide on a time-zero, which can, of course, be the emission time of either of the sources. Thus for each receiver we obtain two pictures, one for each source (in general for n_S sources and n_B receivers there are $(n_S \times n_B)$ possible displays), and the four possible pictures are shown in Fig 8.5. Again one can see a strong cigar shaped return pointing towards the source except in the top right picture which is monostatic. Again there are ambiguous beam returns for receiver 2 but not for receiver 3, for which the target happens to be endfire. The left hand two frames show the same scenario as in Fig 8.3, still taking the first source as our time reference, but with the addition of serious interference from the additional source (the semi-elliptical red shape in the top frame, and the echo in the bottom right corner of the bottom left frame) that could easily mask desired target echoes. Similar effects are seen in the two right hand frames where we now synchronize to source 2. Because source 2 is at the same location as receiver 2 the top right frame is essentially a monostatic sonar with minor interference at bottom left. The bottom right frame escapes interference purely by virtue of the longer delay of source 1 after source 2. If we were to

look further out in range we would find similar elliptical ghosts and repeats of the target echoes and, more generally, ghosts of reverberation clutter features. It is stressed that all these features come quite naturally from the (t,ϕ) to (x,y) mapping of the overlaid time-beam angle plots.

9 Conclusions

SUPREMO is a versatile new multistatic sonar performance model capable of producing time-bearing or geo-plots of reverberation and targets. This report justifies the algorithms currently used in SUPREMO and some future ones such as ambient noise and Doppler. Of particular interest were the modularity and novel approaches taken to some well-known modelling problems. Separate displays are available for each source-receiver pair. These displays are designed to show operators, operational analysts and trials planners where their best detection opportunities are likely to be. The model allows for insertion of any propagation model that can provide intensity, delay time and vertical angle. A design philosophy is to allow the user to control the trade-off between computer speed and fidelity in a simple fashion. Multipath arrivals are considered on the in-going and out-going paths to each scatterer. Computation time is optimised by using Delaunay triangles to rationalise the distribution of scatterers in time-beam space. A post-processing technique converts the impulse response (ie intensities in finely spaced time-bearing bins) into the response for the given pulse and receiver characteristics. The model can handle arrays in any orientation, source beam patterns, two-dimensional target directionalities, arbitrary numbers of sources and receivers on arbitrary numbers of platforms. Above all, computation is very fast and typically takes seconds rather than hours. A separate User Guide [5] describes the Graphical User Interface (GUI) and how to run the model.

References

[1] Harrison, C.H., Prior, M. Multistatic Reverberation and System Modelling using SUPREMO, BOUNDARY2001 JRP meeting, Halifax, Canada, October 2001.

[2] Harrison, C.H., Prior, M. Baldacci, A. Multistatic sonar system modelling using 'SUPREMO' , Proc. UDT, La Spezia, Italy, June 2002.

[3] Harrison, C.H. Multistatic reverberation and system modelling using SUPREMO, Proc. ECUA, Gdansk, Poland, June 2002.

[4] Prior, M., Harrison, C.H., LePage, K.D. Reverberation modelling using acoustic arrivals derived from mode sums and ray tracing, Proc. ECUA 2002, Gdansk, Poland, June 2002.

[5] Baldacci, A., Harrison, C.H. SUPREMO Prototype Version 0.1 User's Guide, SACLANTCEN M-141.

[6] Westwood, E.K., Vidmar, P.J. Eigenray finding and time series simulation in a layered-bottom ocean. *Journal of the Acoustical Society of America*, **81**, 1987:912-934.

[7] Bini-Verona, F., Nielsen, P.L., and Jensen, F.B. PROSIM broadband normal-mode model. User's guide, SACLANTCEN SM-358.

[8] Gradshteyn, I.S., Ryzhik, I.M. Table of Integrals, Series, and Products, Academic Press, 1980.

[9] Morse, P.M., Feshbach, H. Methods of theoretical physics, McGraw-Hill, NY, 1953.

[10] Rabiner L.R., Gold, B. Theory and Application of Digital Signal Processing, Prentice-Hall, NJ, 1975.

[11] Harrison, C.H. A comparison of modelled signal and noise time series with measurements, OCEANS'98, Nice, September 1998.

[12] Using MATLAB. Math Works, Inc., 1996.

Annex A: What is Intensity?

It is often glibly stated that

$$I = p^2 / \rho c \tag{A1}$$

without saying whether this formula is in the time domain or the frequency domain. The answer matters because:

If it's time domain then:

$$i(t) = \left| p(t)^2 \right| / \rho c \quad \textit{Therefore} \quad I(\omega) = \int P^*(\omega') P(\omega + \omega')\, d\omega / \rho c$$

If it's frequency domain:

$$I(\omega) = \left| P(\omega)^2 \right| / \rho c \quad \textit{Therefore} \quad i(t) = \int p^*(t') p(t + t')\, dt / \rho c$$

They cannot both be correct because they contradict each other. Whether we're talking about acoustic intensity or electrical power the same reasoning applies.

Acoustic intensity is force × particle velocity in the direction of the force, ie the dot product of acoustic pressure and particle velocity ([9] pp 309-312). In this context the 'pressure' and 'particle velocity' are *instantaneous values*, ie values at one time. These values consist of the sum of all Fourier components. Therefore

$$\begin{aligned} i(t) &= p(t) v^*(t) = \left| p(t)^2 \right| / \rho c \\ &= \int P(\omega) \exp(-i\omega t)\, d\omega \left(\int V(\omega) \exp(-i\omega t)\, d\omega \right)^* / \rho c \\ &= \left| \int P(\omega) \exp(-i\omega t)\, d\omega \right|^2 / \rho c \\ I(\omega) &= \int P^*(\omega') P(\omega + \omega')\, d\omega / \rho c \end{aligned} \tag{A2}$$

Similarly *Electrical Power* w is instantaneous volts times amps, or in terms of volts with impedance Z

$$w(t) = \left| v(t)^2 \right| / Z$$
$$W(\omega) = \int V^*(\omega') V(\omega + \omega')\, d\omega / Z \qquad \text{(A3)}$$

This says that stating the pressure or voltage at one time defines the intensity at that time. However stating the pressure or voltage at one frequency *does not define the intensity* at that frequency because there may be cross-terms from other frequencies.

Annex B: Doppler

It is clear that you can't change the order of convolutions (as in Eqs 3.4,3.5,3.6) if we're interested in Doppler because the Doppler shift that happens somewhere inside q alters all the frequencies before they get into p_R. Thus the match of p_R and p_S is affected, whereas it wouldn't have been if we had correlated the two pulses before hand. Another way of stating this is that Doppler results from a time-varying filter. However to prove that the output of a filter is the convolution of its input and response we need to invoke linear-time-invariance of the filter ([10], p13). Therefore we cannot use simple convolution with a Doppler system. There are messy ways round it though [11].

Of course, we can still go ahead and calculate frequency shifts and record an array $RL(\phi, t, \Delta f)$, but this is on the implicit assumption that the correlation is unchanged. In fact for a well-designed FM sweep this is a good assumption; but the ambiguity diagram says there is a trade-off between Δf and Δt *because* of this invariance of match to Doppler!

Annex C: Taking frequency averages of propagation loss

How do we use transmission loss, calculated at a set of frequencies, in a time domain model?

To help establish the correct procedure, consider three cases, all one-way propagation.

• *Coherent wave solution for many separate frequencies:*

Suppose we run a model such as PROSIM at many equally spaced frequencies (ie we obtain Q(ω) numerically) and then add the solutions coherently, ie we form

$$q(t) = \frac{1}{2\pi} \int Q(\omega) \exp(-i\omega t)\, d\omega$$

q(t) consists of multiple arrivals each of finite amplitude and blurred shape. Thus the impulse response, in practice, does not contain any delta functions because Q(ω) inevitably dies off at high and low frequencies. There are no delta functions in the response because the source is not a delta function.

• *Spherical spreading "time t = r/c"*

Now we repeat the exercise with an ideal frequency-independent environment (we're allowed phases, but not amplitude variations!) where

$$Q(\omega) = \exp(ikr)/r$$

The impulse response has infinite amplitude and

$$q(t) = \delta(t - r/c)/r$$

but, of course, $\int q(t)dt = 1/r$.

• *Sonar eq for "time t = r/c"*

In the sonar equation, typically we have slowly varying frequency-dependent terms.

We deal with them in the same way as 1/r as above. But how come we never have any delta functions to deal with when we use the sonar equation in the time domain? For instance, we would imagine a pulse arriving at time t = r/c with *finite* amplitude 1/r.

To demonstrate this, think of a gaussian pulse

$$p(t) = \exp(i\omega_o t)\exp(-t^2 / t_p{}^2)$$

whose FT is $P(\omega) = \sqrt{\pi}\ t_p \exp(-t_p{}^2(\omega - \omega_o)^2 / 4)$.

With or without Parseval's theorem we have

$$\int |p(t)|^2 dt = \frac{1}{2\pi}\int |P(\omega)|^2 d\omega = \sqrt{\frac{\pi}{2}}\ t_p$$

From the definition of P we could just as easily have started with a bandwidth $\omega_b = 2/t_p$.

What we would do in the sonar equation is compare the received pulse $p \otimes q$ with the transmitted pulse p

$$\begin{aligned} p \otimes q &= \int \delta(t'-r/c)/r \times \exp(i\omega_o(t-t'))\exp(-(t-t')^2 / t_p{}^2)\, dt' \\ &= \frac{1}{r}\exp(i\omega_o(t-r/c))\exp(-(t-r/c)^2 / t_p{}^2) \end{aligned}$$

Comparing peak amplitudes we get a phase shift and a factor of 1/r. Alternatively we compare power time integrals, ie $\int |p \otimes q|^2\ dt$ with $\int |p|^2\ dt$ to obtain $1/r^2$.

These three cases suggest that in SUPREMO when we calculate Q(ω) with a propagation program we should integrate it over the passband P(ω) associated with the source and receiver pulses and any additional receiver filtering. However SUPREMO already does a genuine convolution with the pulse in `postproc`, so to avoid double counting we need to use a frequency average transmission loss defined as follows

$$\int |P(\omega)|^2 |Q(\omega)|^2 d\omega \Big/ \int |P(\omega)|^2 d\omega$$

Annex D: Taking frequency averages of beam patterns

In `postproc2.m` there are two smudging effects, one represents the finite pulse length, the other represents the finite beam width. Because the beam pattern is a function of frequency we have to average it somehow. The "signal" for postproc is actually a power so convolving with a power beam pattern is a linear process. There is therefore no difference between performing postprocessing with the correct beampattern separately at each frequency, and performing postprocessing once outside the frequency loop with the (weighted) average beam pattern. The weighted average is carried out in exactly the same way as the average for propagation loss (Annex C).

Document Data Sheet

Security Classification		Project No.
NATO UNCLASSIFIED		
Document Serial No.	**Date of Issue**	**Total Pages**
SM-396	November 2002	41pp.

Author(s)

Harrison, C.H.

Title

SUPREMO: A Multistatic Sonar Performance Model

Abstract

Multistatic sonars introduce new problems to the modeller such as mutual interference from multiple sources and variable target aspect in a background of range-dependent bistatic reverberation. A multistatic sonar performance model, SUPREMO, has been developed at SACLANTCEN. Its capability includes propagation, target echoes, reverberation and noise, plotted as a function of delay time and bearing, and in equivalent map projection.
A modular propagation section makes it possible to separate the difficulties of propagation modelling (ray/ mode, range-dependence, etc) from those of scatter-area-accounting in delay-time and bearing (computational efficiency, bottom slope, scattering law, etc). Several novel trade-offs of efficiency versus fidelity are discussed.

Special attention is paid to the problems of interference from multiple sources firing in sequence, target aspect dependence from multiple receivers, mixed FM and CW, mismatched source and receivers, multiple displays (one for each bistatic pair), and how to combine the displayed information.

Keywords

Multistatics – reverberation – target echo – propagation – sonar performance – real-time trials planning

Issuing Organization

North Atlantic Treaty Organization
SACLANT Undersea Research Centre
Viale San Bartolomeo 400, 19138 La Spezia, Italy

*[From N. America: SACLANTCEN
(New York) APO AE 09613]*

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

E-mail: library@saclantc.nato.int