Copy No. _____

Defence Research and Development Canada | Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

BellhopDRDC Users Guide, Version 3

Covering Transmission Loss, Ray Tracing, Bottom Loss and Surface Loss

Dr. Diana McCammon
McCammon Acoustical Consulting

McCammon Acoustical Consulting
475 Baseline Road
Waterville, NS
B0P 1V0

Contract Number: W7707-053203

Contract Scientific Authority: Dr. WA Roger 902-426-3100 x292

Defence R&D Canada – Atlantic

Contract Report
DRDC Atlantic CR 2006-067
December 2006

Canada

This page intentionally left blank.

BellhopDRDC Users Guide, Version 3

Covering Transmission Loss, Ray Tracing, Bottom Loss and Surface Loss

Dr. Diana McCammon
McCammon Acoustical Consulting

McCammon Acoustical Consulting
475 Baseline Road
Waterville NS B0P 1V0
Canada

Contract number: W7707-053203
Contract Scientific Authority: Dr. WA Roger 902-426-3100 x292

Defence R&D Canada – Atlantic

Contract Report

DRDC Atlantic CR 2006-067

December 2006

Principal Author

Dr. Diana McCammon

Approved by

David Hazen

H/TD

Approved for release by

K. Foster

DRP

Abstract

Over the most recent series of contracts the acoustic prediction model called Bellhop has been streamlined to more closely fit the requirements of DRDC Atlantic's Environment Modeling Manager. The Bellhop Fortran code has been streamlined by removing choices of interest in scientific research but less necessary in an operational system. The input data and file formats have been altered to satisfy the requirements of the controlling programs within the Environment Modeling Manager, and additional output capabilities for bistatic reverberation and active signal excess have been added. The program has been configured into two executables to run for passive or active prediction. This document provides a user's guide to the running of Bellhop, and describes some plotting routines available for viewing the prediction results.

Résumé

Dans le cadre de la dernière série de contrats, on a rationalisé le modèle prédictif de champs sonores appelé Bellhop afin de mieux satisfaire aux exigences de l'Environment Modeling Manager de Recherche et développement pour la défense Canada – Atlantique (RDDC). On a simplifié le code Fortran de Bellhop en supprimant des choix présentant un intérêt pour la recherche scientifique, mais moins nécessaires dans un système opérationnel. On a modifié les formats des données d'entrée et des fichiers afin de satisfaire aux exigences des programmes de contrôle d'Environment Modeling Manager et ajouté des capacités de sortie supplémentaires pour la réverbération bistatique et l'excès de signaux actifs. Le programme a été configuré en deux exécutables pour la prédiction passive ou active. Ce document constitue un guide d'utilisation de Bellhop et décrit certaines routines de traçage permettant de visualiser les résultats des prédictions.

This page intentionally left blank.

Executive summary

BellhopDRDC Users Guide, Version 3

Dr. Diana McCammon DRDC Atlantic CR 2006-067; Defence R&D Canada – Atlantic; December 2006

Introduction

Over the past few years the acoustic prediction program called Bellhop has been enhanced to accommodate both passive and active environmental prediction. This application computes the acoustic fields via beam tracing, and can handle variations in sound speed profile, bottom loss, and bathymetry. The calculated results include transmission loss, based on coherent, semi-coherent, or incoherent summation, and an arrival structure or SALT (Sound Angle, Level and delay Time) table.

Results

This document is a User Guide to the Bellhop program. It details the input requirements, commands, and options to successful run the engine, and provides a number of example input configuration files. Plots extracted from sample output files are also shown.

Significance

Tactical oceanography is a critical aspect of Underwater Warfare, providing predictions of sonar performance and detection probabilities. The Bellhop acoustic prediction program serves as an underlying engine to operator requests, tactical decision aids, and combat system algorithms during at-sea operations against subsurface targets. This document assists programmers involved in algorithm development to realize the full potential of the Bellhop engine.

Future plans

Future versions of Bellhop will include more capability in the prediction of both passive and active sonar performance, particularly in littoral waters.

Sommaire

Guide d'utilisation de BellhopDRDC, version 3

Dr. Diana McCammon, DRDC – Atlantique, DCD 2006-067; Recherche et développement pour la défense Canada – Atlantique, decembre 2006

Introduction

Au cours des quelques dernières années, on a amélioré le programme de prédiction de champs sonores appelé Bellhop afin de permettre la prédiction environnementale passive et active. Cette application calcule les champs sonores par traçage de faisceaux et peut gérer les variations dans le profil de vitesse du son, les pertes au fond et la bathymétrie. Les résultats calculés comprennent l'affaiblissement acoustique, basé sur la sommation cohérente, semi-cohérente ou incohérente, et une structure d'arrivée ou table SALT (*Sound Angle, Level and delay Time*).

Résultats

Ce document est un guide d'utilisation du programme Bellhop. Il décrit de façon détaillée les exigences d'entrée, les commandes et les options permettant d'exécuter avec succès le programme, et présente plusieurs exemples de fichiers de configuration d'entrée. Des tracés extraits d'exemples de fichiers de sortie sont également présentés.

Importance

L'océanographie tactique est un aspect crucial de la guerre sous-marine, qui fournit des prédictions sur la performance des sonars et des probabilités de détection. Le programme de prédiction de champs sonores Bellhop sert de moteur sous-jacent aux demandes de l'opérateur, aux aides à la décision tactique et aux algorithmes de systèmes de combat durant les opérations en mer contre des cibles sous-marines. Ce document vise à aider les programmeurs chargés du développement d'algorithmes à réaliser le plein potentiel du moteur Bellhop.

Plan futurs

Les futures versions de Bellhop comprendront des capacités supplémentaires pour la prédiction de la performance des sonars passifs et actifs, en particulier dans les eaux littorales.

Table of contents

List of figures

List of tables

This page intentionally left blank.

1. Introduction

Bellhop is a computer program created by Dr. Michael Porter that computes acoustic fields in oceanic environments via Gaussian beam tracing. The environment consists of an ocean that may have range variations in the sound speed profile, the bottom loss, and the bathymetry. The outputs include transmission loss (coherent, semi-coherent, and incoherent), and an arrival structure or SALT (Sound Angle, Level and delay Time) table which can be used for active predictions.

This users guide describes the two versions of Bellhop named BellhopDRDC_ray_TL_v3 and BellhopDRDC_active_v3 that were created for use with the Environment Modeling Manager (EMM) at the Defence R&D Canada – Atlantic (DRDC Atlantic) laboratory.

Changes in version 3:

The changes between this third version and the previous versions include:

1. The creation of two models from the original Bellhop for passive and active. The first, BellhopDRDC_ray_TL_v3 computes passive transmission loss or ray trace files at the users choice. The second, BellhopDRDC_active_v3 computes SALT tables, signal time series, surface and bottom reverberation time series using a bottom conforming output, and active signal excess.
2. The inclusion of beam patterns in both models.
3. The choice of scattering strength models for the active model. The choice of surface and bottom loss models with both models.
4. The dynamic allocation of most array sizes, removing most input size restrictions.
5. The incorporation of changes from Dr. Porter's Web version of Bellhop dated Jan 2006.

The Fortran coding approaches are similar between the two models. They consist of a frontend program that reads the input files and writes the output files, and a subroutine named BellhopDRDC. This structure was used to enable repeated calls to the subroutine Bellhop from within frontend, for looping calculations over source depth, frequency, or bearing for example. It is anticipated that the user may rewrite or replace the frontend algorithm to suit his application.

Also included in this users guide is a simplified program to compute surface and bottom loss for separate analysis. Finally, included in this users guide are examples of some IDL plot routines for TL vs range, (full field or single depth), ray tracing, arrival angle, surface loss, bottom loss, reverberation and signal excess.

2. BellhopDRDC_ray_TL_v3

The BellhopDRDC_ray_TL_v3 model is intended for passive predictions of ray paths and transmission loss. This model consists of four Fortran source files and their subroutines:

1. datamod_ray_TL_v3.f90 – module with data array declarations

2. refcomod_ray_TL_v3.f90 – module with reflection coefficient array declarations and some loss models

   CALCbotRC – compute bottom reflection coefficients using MGS NAVOCEANO routine
   CALCtopRC – compute surface reflection coefficient using either Modified Eckart or Beckmann-Spezzichino
   BOTT_NEW – MGS bottom loss function
   SURF_NEW – Surface loss function using Bechmann-Spezzichino
   LFSOPN – Low Frequency Open Ocean Surface loss using Modified Eckart

3. frontend_ray_TL_v3.f90

   frontend_ray_TL_v3 – main program
   clean_up – deallocate ray structure arrays
   Raywrite – write out to ascii rays.txt the ray trace path information
   READIN_v3 – reads file runinput.inp and allocates and initializes arrays for range and receiver depth
   READBTY_v3 – reads file bathy.inp and allocates arrays for bathymetry
   READSVP_v3 – reads file speed.inp and allocates arrays for sound speed
   READBOTLOSS_v3 – reads file bottomloss.inp and allocates arrays for which ever bottom type was specified
   READBPATTERNS_v3 – reads file beampattern.inp for sensor beam pattern, allocates arrays and converts loss to pressure coefficient

4. bellhopDRDC_ray_TL_v3.f90

   BellhopDRDC_ray_TL_v3 – beginning of bellhop algorithm- initializes arrays, calls ray trace and calls Grab style TL computation
   Trace – traces a ray for each launch angle
   Step – takes a single step along the ray path
   Reflect – changes ray direction and computes amplitude and phase at reflection. The geoacoustic bottom loss is embedded in this subroutine.
   REFCO – interpolates for reflection coefficients from table if needed
   INFLUGRB – Gaussian beam contribution to complex pressure for TL
   TMP_SPP – function to convert temperature to sound speed using Leroy's equation
   CLIN – linear interpolation of sound speed data with depth
   Smoother – Savitsky-Golay smoothing filter for coherent TL
   Thorpe – Thorpe attenuation
   CRCI – converts real wave speed and attenuation to a single complex wave speed
   ERROUT – outputs error messages

The two module files, datamod_ray_TL_v3.f90 and refcomod_ray_TL_v3.f90, contain the data arrays and declarations, and must be compiled first. The executable is named BellhopDRDC_ray_TL_v3.exe

To run the program, place the executable BellhopDRDC_ray_TL_v3.exe in your working directory or on your path. Place the five input files listed below in your working directory. Then click on the .exe icon or use the windows start/run command. If programming in IDL, the spawn command can be used to run the executable. For example, the command to run this in IDL is: spawn, 'BellhopDRDC_ray_TL_v3.exe', result, /noshell .

2.1 Input files

There are five input files: runinput.inp, speed.inp, bottomloss.inp, bathy.inp and beampattern.inp. The formats are free field, so the values on each row do not occupy specific column positions, but only need be separated by a space.

2.1.1 Runinput.inp

This file contains scenario and runtime choices, as defined in Table 1 and illustrated in Figure 1. In this table, the following alphabetic choices are defined:

‘X’ = the run choice option

- 'C' = Coherent transmission loss in output file CTL.txt
- 'S' = Semi-coherent transmission loss in output file STL.txt
- 'I' = Incoherent transmission loss in output file ITL.txt
- ‘R’= Ray trace path information in output file rays.txt

‘S’ = surface loss model choice

- ‘B’ = Beckmann Spezzichino surface loss
- ‘E’ = Modified Eckart low frequency open ocean surface loss = default model

(Note that the bottom loss model is chosen in the bottomloss.inp file)

Table 1. runinput.inp file structure

Line #, entry		***Notes***
1.	title	up to 70 characters enclosed in single quotes
2.	frequency	Hz
3.	source depth	Meters
4.	number of receiver depths	
5.	top and bottom of receiver depth array	Meters- note: needs slash at end value to denote an array- a single value can also be used
6.	Range step for output; longest range	Meters; Kilometres
7.	wind speed; surface loss model choice	Knots; ‘S’ ={B,E}
8.	Run Choice options	‘X’ ={C,S,I,R}
9.	Internal step size; number of rays; start angle; stop angle; kill-after-bounce number	Default value = -1 Internal step size in m; angles in degrees; negative angles first. Default is -45 to 45 deg, and 100 bounces For ray tracing, the number of rays and start and stop angle should be selected by the user. For transmission loss, these should be defaulted to -1.
10	Range smoothing flag	Meters, Default = -1, no smoothing Smoothing only affects the ‘C’ coherent TL

```
‘Emerald basin toward Sambro Bank’    !title
1200.0           !frequency (Hz)
21.              !source depth, m
2                !number of receiver depths
21. 80. /        !top and bottom of receiver depth array, or whole array, ** needs the slash
750. 50.         !range step (m) and maximum range (km)
 15.0  'B'       !windspeed(kts), surface loss model {B,E, }
'I'              !run choice {I,S,C,R}
-1 -1 -1 -1 -1   !defaults, step size (m), number of rays, start and stop angles, Kill-after-bounce
-1               !smoothing default (-1)=off,  1=on
```

Figure 1. Sample runinput.inp file.

2.1.2 Speed.inp

This file contains sound speed profiles in depth and range. The format is defined in Table 2 and an example shown in Figure 2.

Table 2. speed.inp file structure

Line #, entry		***Notes***
1	Number of range dependent profiles	
2	Range to profile; number of points in that specific profile, n	km
3 to n+3	Depth; speed or temperature	M; m/sec or ºC
	Repeat from 2 for each profile	

Note: there should always be a point at the surface and at or below the deepest bathymetry point.

```
  1                    !# range dependent profiles
 0.   18               !range(km),   #points per profile
 0.     1498.0         !depth(m)   speed(m/sec)
30.0   1499.2
35.0   1491.7
40.0   1483.8
45.0   1475.4
50.0   1466.5
75.0   1468.2
80.0   1470.0
90.0   1473.5
95.0   1475.3
100.0  1477.0
125.0  1479.6
150.0  1482.1
175.0  1484.6
200.0  1488.4
225.0  1489.5
250.0  1490.6
300.0  1490.5
```

Figure 2. Sample speed.inp file.

2.1.3 Bottomloss.inp

This file contains the range dependent bottom loss descriptions. The format is defined in Table 3 and an example shown in Figure 3.

Table 3. bottomloss.inp file structure

Line #, entry		***Notes***
1	Bottom treatment option; attenuation units	‘XY’: ‘X’= {M,A,T} ‘Y’= {F,M,W,N}
2	number of range dependent bottom sets, n	
3 to n+3	If ‘X’=‘M’: range; province number If ’X’=’A’: range; c1; rho1; atten1; h1; c2; rho2; atten2 If ‘X’=’T’: range; # of table rows Angle; reflection coefficient; phase	Km; MGS province number Km; m/sec; g/cc; units of ‘Y’; m; m/sec; g/cc; units of ‘Y’ Km; number of rows Degrees; decimal fraction; degrees

In this table, the following are defined:

‘X’ = the bottom treatment option
- 'M' = MGS or HFBL provinces
- 'A' = Geoacoustic fluid layers (no shear)
- 'T' = Read in table of pressure reflection coefficients and phases as a function of grazing angle

‘Y’ = The attenuation units which are used in the geoacoustic layers only, choices are
- 'F' = dB/(m kHz)
- 'M' = dB/m
- 'W' = dB/wavelength
- 'N' = nepers/m

```
 'M'              ! Bottom option for MGS
 3                ! number of range dependent bottom provinces
 0.   4           ! range (km),    province number
 10.  2
 50.  8
```

```
  'AF'            !A=geoacoustic, F= dB/m kHz
 7                !number of bottom regions
  0.0  1453.  1.41   0.038  10.  1557.  1.73   0.156
 18.5  1547.  1.72   0.158  20.  1880.  2.175  0.085
 22.2  1807.  2.175  0.131  30.  3500.  2.60   0.020
 25.9  1630.  2.00   0.157   4.  1880.  2.175  0.085
 27.8  1772.  2.11   0.142  10.  1880.  2.175  0.085
 44.4  1807.  2.175  0.131  30.  3500.  2.60   0.020
 50.0  1453.  1.41   0.038  10.  1557.  1.73   0.156
!range   c1    rho1   atten1  depth  c2   rho2    atten2
```

```
 'T'              ! Bottom option for table of reflection coefficients vs angle
  1               ! number of sets of tables
 0.  5            ! range (km), number of entries in table
  0.  1.0  0.0    ! angle (deg), reflection coef fraction, phase (deg)
 10. 0.8  0.0
 30. 0.7  0.0
 50. 0.5  0.0
 90. 0.5  0.0
```

Figure 3. Three samples of bottomloss.inp files.

2.1.4 Bathy.inp

This file contains the bathymetry. The format is defined in Table 4 and an example shown in Figure 4.

Table 4. bathy.inp file structure

Line #, entry		***Notes***
1	Number of bathymetry points, n	
2 to n+2	Range; depth	Km; m

Note: needs a point at zero range

```
   12                          !number of bathymetry points
  0.0   238                    !range(km), bottom depth (m)
  5.5   256
 18.5   238
 22.2   219
 24.1   183
 25.9   165
 27.8   110
 42.6   128
 44.4   146
 46.3   165
 47.2   183
 55.5   219
```

Figure 4. Sample bathy.inp file.

2.1.5 Beampattern.inp

This file contains the receiver vertical beam pattern in dB. The format is defined in Table 5 and examples shown in Figure 5 and Figure 6.

Table 5. Beampattern file structure

Line #, entry		***Notes***
1	Number of vertical angles, n	
2 to n+2	Angle; loss	Deg; dB

```
3                              !number of angles
-90.  0.0                      !angle(deg); loss(dB)
  0.  0.0
 90.  0.0
```

Figure 5. Sample beampattern.inp for an omni-directional beam.

```
37                          !number of transmitter angles
-90.0   51.4                !angle(deg), loss dB
-80.0   21.2
-70.0   28.5
-60.0   19.85
-50.0   20.55
-45.0   25.3
-40.0   19.0
-35.0   14.0
-30.0   13.66
-25.0   20.0
-20.0   18.44
-17.5   11.69
-15.0   7.72
-12.5   5.00
-10.0   3.05
-7.5    1.66
-5.0    0.721
-2.5    0.178
0.0     0.0
 2.5    0.178
 5.0    0.721
 ...
```

Figure 6. Sample beampattern.inp file for a transmitter beam.

2.2 Output files

There are five possible output files from BellhopDRDC_ray_TL_v3. The computed data is written to .txt files in ASCII, depending on the runtime choice made in the input file runinput.inp.

ITL.txt created by run choice 'I'
STL.txt created by run choice 'S'
CTL.txt created by run choice 'C'
rays.txt created by run choice 'R'
bellhop.log

2.2.1 CTL.txt, ITL.txt or STL.txt

This file contains the transmission loss (either coherent, semi-coherent or incoherent, depending on the choice made in runinput.inp). At the top, it lists the run title, frequency and source depth. The next line contains the number of ranges and number of receiver depths. Following this are listed the range array in km, then the receiver depth array in m, then transmission loss in dB by range and receiver depth. An example output file is shown in Figure 7 and a plot using this data is illustrated in Figure 8.

```
BELLHOP- Emerald basin toward Sambro Bank

 1200Hz   21.0m source depth
        66          2
 0.7500000      1.500000      2.250000      3.000000      3.750000
  4.500000      5.250000      6.000000      6.750000      7.500000
  8.250000      9.000000      9.750000      10.50000      11.25000
  12.00000      12.75000      13.50000      14.25000      15.00000
     . . .
  45.75000      46.50000      47.25000      48.00000      48.75000
  49.50000
  21.00000      80.00000
  56.04723      60.83149      61.22410      61.61346      67.61076
  66.07870      66.25555      70.31084      73.15980      69.96243
  71.60814      75.70319      73.49827      73.55996      76.92455
     . . .
```

Figure 7. Portion of an ITL.txt output.

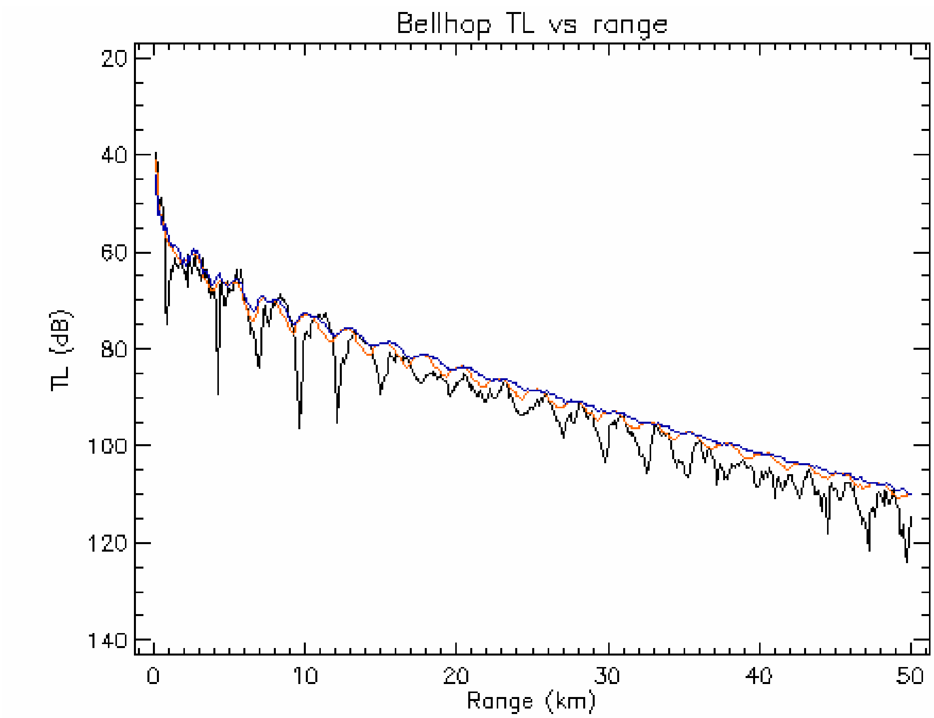


Figure 8. Example transmission loss plot for 20m receiver. Black is coherent, CTL.txt. Red is semi-coherent, STL.txt and blue is incoherent, ITl.txt, using 500 range points.

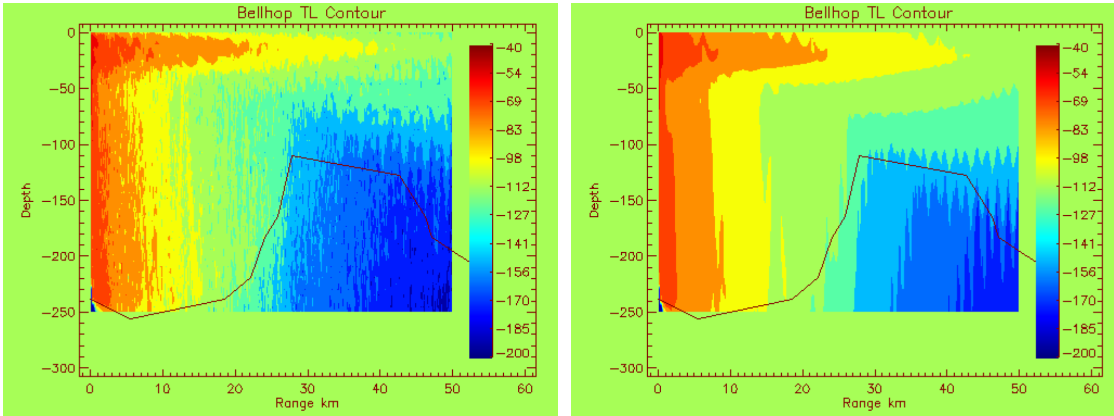


Figure 9. Left: example of full field plot of CTL.txt, (coherent calculation) which was computed using 50 receiver depths from 0 to 250m. The bathymetry is plotted as a line along the bottom. Right: full field plot of ITL.txt (incoherent calculation).

Figure 9 displays a good example of a potential pitfall in using this range dependent model. Note that in the figure, there are places where the field extends below the bathymetry, since the receiver array was defined to 250m to cover the deeper part of the water, but the bathymetry then rises to 110m. The portion of the field below the bathymetry is not a true representation of the acoustic field there. The loss generated by Bellhop on reflection from the bottom into the water column is correct, however the field shown within the bottom does not have the right level. It is an artifact of the Gaussian beam representation in Bellhop. It should be ignored or blanked out in the figure, and in all other Bellhop applications, care should be exercised that the user is only working with transmission loss values from those receivers positioned above the bathymetry.

2.2.2 Rays.txt

The output file named Rays.txt contains ray tracing information. Its structure is to echo some of the input choices in the first few lines. The number of rays being traced is listed (in the case shown below it is 6). Then in a loop over the number of rays, each ray is described by the launch angle (-3.0) and number of steps or points in the trace (7481). Finally, the [r,z] coordinates, ray angle, delay time, and number of surface and bottom bounces of each ray are listed for each step. Both r and z are given in metres, angle is in degrees, and time is given in seconds.

Figure 10 shows a portion of the rays.txt listing for the 21m source, and it demonstrates an anomaly that always occurs in Bellhop ray traces. That is, the point of reflection from the surface or bottom is always repeated, as shown at 366.4238 m. On the second repeat, the angle is zero indicating a change of direction.

The rays.txt output can be plotted with the bathymetry, as shown in Figure 11. The case shown was computed using 6 rays from -3º to +3º, with a 21m source depth so that the figure would correspond directly to the full field transmission loss plot in Figure 9.

```
BELLHOP- Emerald basin toward Sanbro Bank

   1200.0Hz      21.0m source depth
Kill Trace after   100 bottom bounces
         6
 -3.000000          7481
 0.0000E+00   21.00000          -3.000000        0.0000000E+00     0      0
  11.88359     20.37532          -3.018170         7.9395389E-03     0      0
  23.76698     19.74687         -3.036340         1.5879210E-02     0      0
  35.65018     19.11465         -3.054511         2.3819016E-02     0      0
    . . .
  344.5391     1.354395          -3.526937        0.2303035        0      0
  356.4165     0.6204501         -3.545108        0.2382472        0      0
  366.4096     5.1173470E-06    -3.560396        0.2449310        0      0
  366.4238    -8.7583682E-04    -3.560417        0.2449404        0      0
  366.4238    -8.7583682E-04    0.00000E+00   0.2449404        1      0
  378.3009     0.7362430          3.542247        0.2528843        1      0
   . . .
```

Figure 10. Portion of a rays.txt output.

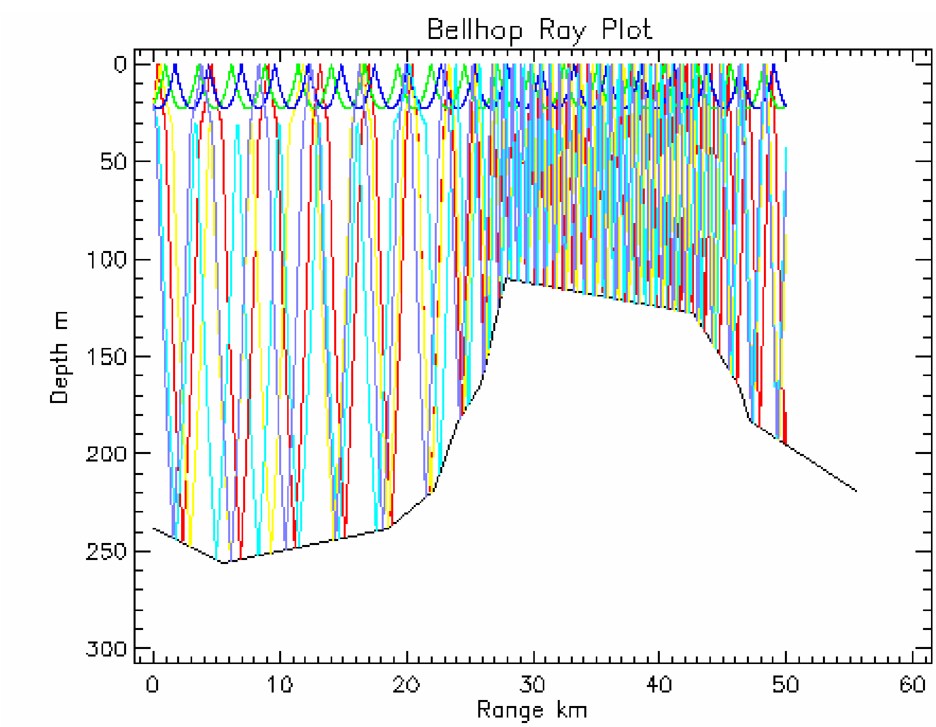


Figure 11. Plot of rays.txt for a 21m source showing the reflections from the uneven bathymetry.

2.2.3 Bellhop.log

This file contains a log of the runtime statements generated in any run. Some inputs are echoed, and any warnings or errors generated by the Bellhop code are listed here. An example is shown in Figure 12.

```
BELLHOP- Emerald basin toward Sambro Bank
Frequency=    1200.000
 Source depth=    21.00000
 Number of receiver depths=            2
 range step(m) =    750.0000      Maximum range(km)=    50.00000
 Wind speed (kts)=    15.00000
 Beckman-Spezzichino surface loss
 Runchoice= I
 No range smoothing
 User input Receiver depth array   21.00000       80.00000
 Number of sound speed profiles=            1
   Range(km)=  0.0000000E+00 #points per profile=           18
           1           1 Depth(m)=   0.0000000E+00 Speed/Temp=    1498.000
           1           2 Depth(m)=    30.00000     Speed/Temp=    1499.200
                                                 . . .
```

Figure 12. Sample portion of bellhop.log.

2.3 Plot routines

Several IDL plot routines have been prepared to provide a simple graphic representation of the output products from BellhopDRDC. These should be freely altered to suit the users' data and output requirements.

Read_tl_plot_loss.pro: Routine to read each of the xTL.txt output files, and then plot the transmission loss against range, as shown in Figure 8. The user will be asked to enter the receiver depth. If it does not exactly match one of the computed depths, the plot routine will choose the next closest depth. Presently the plot routine is set to open each of the three xTL.txt files and over plot them all in color.

Read_tl_plot_field.pro: Routine to read the xTL.txt output and the bathy.inp file and plot the full field as shown in Figure 9. To obtain the best picture, the range of levels of loss that will be shown should be adjusted by the user in the variable Lev.

Read_rays_plot_trace.pro: Routine to read the output in Rays.txt and the bathymetry in bathy.inp, and produce a ray trace figure as shown in Figure 11.

3. BellhopDRDC_active_v3

The BellhopDRDC_active_v3 model is intended for active predictions of bistatic signal time series, bistatic reverberation and active signal excess using SALT (Sound Angle, Level and Time) tables produced by the incoherent output from Bellhop. This model consists of eleven Fortran source files and their subroutines:

1. datamod_active_v3.f90 – module of data array declarations

2. refcomod_active_v3.f90 – module of reflection coefficient array declarations

   CALCbotRC – computes bottom reflection coefficients using MGS NAVOCEANO routine
   CALCtopRC – computes surface reflection coefficient using either Modified Eckart or Beckmann-Spezzichino
   BOTT_NEW – MGS bottom loss function
   SURF_NEW – Surface loss function using Bechmann-Spezzichino
   LFSOPN – Low Frequency Open Ocean Surface loss using Modified Eckart

3. saltmod_active.f90 – module with SALT table array allocation declarations

4. SEmod_active.f90 – module with signal excess (SE) input variable allocation declarations

5. frontend_active_v3.f90

   frontend_active_v3 – main program
   Setdefaults – assigns default inputs for active applications

6. readinput_active.f90

   readinput_active – reads input files for SE, speed, bathy, beam patterns
   READBOTLOSS_S – reads bottom loss and allocates arrays for which ever bottom type was specified
   READreverb – read user input reverberation table
   CALCreverb - Rough estimate of reverb in dB using 40logt fall-off

7. bellhopDRDC_active_v3.f90

   BellhopDRDC_active_v3 – beginning of bellhop algorithm- initializes arrays, calls ray trace and calls TL computation- defines extra receiver points on surface and bottom (conforming to bathymetry) for reverberation
   Trace – traces a ray for each launch angle
   Step – takes a single step along the ray path
   Reflect – changes ray direction and computes amplitude and phase at reflection. The geoacoustic bottom loss is embedded in this subroutine.
   REFCO – interpolates for reflection coefficients from table if needed
   INFLUGRB – computes Gaussian beam contribution to complex pressure. For surface and bottom points for reverberation, the last reflection loss is removed. Results are sent to AddArr

TMP_SPP – function to convert temperature to sound speed using Leroy's equation
CLIN – linear interpolation of sound speed data with depth
Smoother – Savitsky-Golay smoothing filter for coherent TL
AddArr – creates arrival SALT table for surface, bottom, and target depths from all sensors and transmitter along all bearings
Thorpe – Thorpe attenuation
CRCI – converts real wave speed and attenuation to a single complex wave speed
ERROUT – outputs error messages

8. reverb_active.f90

    Reverb_active – computes bistatic reverberation from surface and bottom using SALT tables for each sensor and target bearing. Formulas for various surface and bottom scattering strengths are embedded. Output is reverberation time series without source level.

9. signal_active.f90

    Signal_active – computes bistatic signal intensity as a function of time, range and depth along target bearing. Output is signal time series without source level or target strength.
    Envstore – moves range dependent environments from input storage arrays into runtime arrays for each bearing and sensor
    NUWsalt – computes SALT tables for each bearing and sensor using a call to BellhopDRDC_active_v3

10. SE_active.f90

    SE_active – computes signal excess from reverb and signal time series and noise. Source level and target strength are applied. The result is smoothed and saved as a function of range, target depth, bearing and sensor

11. writeoutput_active.f90

    WriteArrival – writes SALT arrival tables for each target bearing and sensor
    Writerevb – writes reverberation time series for each target bearing and sensor
    WriteSE – writes SE for target bearing, target depth, range and sensor
    Writesignal – writes signal time series for each target bearing, target depth, range and sensor
    WriteTL – writes TL from transmitter and from sensor to target vs range for target depth

The four module files, datamod_active_v3.f90 and refcomod_active_v3.f90, saltmod_active.f90 and SEmod_active.f90, contain the data arrays and declarations, and must be compiled first. The executable is named BellhopDRDC_active_v3.exe

To run the program, place the executable BellhopDRDC_active_v3.exe in your working directory or on your path. Place the five input files listed below in your working directory. Then click on the .exe icon or use the windows start/run command. If programming in IDL, the spawn command can be used to run the executable. For example, the command to run this in IDL is: spawn, 'BellhopDRDC_active_v3.exe', result, /noshell .

3.1 Input Files

There are five input files: NUWSEinput.inp, NUWspeed.inp, NUWbottomloss.inp, NUWbathy.inp and NUWbeampattern.inp. The formats are free field, so the values on each row do not occupy specific column positions, but only need to be separated by a space. For active use, the run choice is defaulted to incoherent 'I' in the file frontend, subroutine setdefaults.

3.1.1 NUWSEinput.inp

This file contains the basic scenario and system parameters and scattering strength and surface loss model choices. The format is defined in Table 6 and an example shown in Figure 13.

Table 6. NUWSEinput.inp file structure

Line #, entry		***Notes***
1	Number of receiving sensors, nrd	
2	Noise level at each sensor	Array 1 to nrd, values separated by spaces, units=dB
3	Source level; detection threshold; target strength; system loss; pulse length	dB; dB; dB; dB; seconds
4	Blast arrival time at sensor	Array 1 to nrd, values separated by spaces, units= seconds sign will position the sensor to the right or left of the transmitter as you face the target bearing. Negative=left, positive=right
5	Frequency	Hz
6	Asset depths	Array 1 to nrd+1, values separated by spaces, units=m, extra last point is the transmitter depth
7	Maximum range to target	km
8	Number of target depths; Target depth minimum; Target depth maximum	Program will create an array of depths, units= m, Note: must have both min and max depth, even if the number of targets=1
9	Wind speed; surface loss model choice 'S'	kts; 'S'={B,E}, default=E
10	Surface scattering strength model choice 'SS'	'SS'={OE, CH}, default=CH
11	Bottom scattering strength model choice 'SB'; Mackenzie coefficient; Normal incidence bottom loss; facet width or RMS slope	'SB'={EC, LB, OM}, default=LB; dB, used with all choices; dB, used with 'EC'; degrees, used with 'EC'

In Table 6, the model choices are:

- ‘S’ = surface loss model choice
  - ‘B’ = Beckmann Spezzichino surface loss
  - ‘E’ = Modified Eckart low frequency open ocean surface loss = default model
  - (Note that the bottom loss model is chosen in the NUWbottomloss.inp file)
- ‘SS’ = surface scattering strength model choice
  - ‘OE’ = Ogden-Erskine surface scattering strength- a combination Chapman Harris with low wind speed algorithms
  - ‘CH’ = Chapman Harris surface scattering strength = default model
- ‘SB’ = bottom scattering strength model choice
  - ‘EC’ = Ellis and Crowe= Lambert’s rule with a high angle facet scattering term
  - ‘LB’ = Lambert’s Rule with Mackenzie Coefficient= default model
  - ‘OM’ = Omni-directional Rule with Mackenzie Coefficient

```
1                            !Number of sensors
70.8                         !Noise level by sensor, dB
230.0  0.0  0.0  0.0  1.0    !SL, DT, TS, Syslos, in dB, pulse length in sec
-12.34                       !blasttime to sensor, right = +  left = -
1200.                        !freq
18.3  21.                    !asset depth array, including transmitter
30.                          !maximum range to target locations   km
2  21.0  80.0                !number of target depths, dmin, dmax
15.        'B'               !wind speed in knots,  surface loss model
'CH'                         !Surface scattering strength model
'OM' 21. 5. 10.               !Bottom scattering strength model, lambert coef(dB), normal
                                     !incidence bottom loss (dB), facet width (deg)
```

Figure 13. Example of NUWSEinput.inp file.

3.1.2 NUWbathy.inp

This file specifies the set of radial bearings and bathymetry, with the radials being common for all receivers and the transmitter. The first line defines the number of radials (*nr*) contained in the file. Line 2 specifies the first radial for the first sensor (as a bearing relative to the line connecting transmitter and receiver), the number of range/bathymetry values (*nd*) for this sensor, and a Boolean (0 or 1) indicating whether the current radial points at the target. Next are *n* lines of range (in km) and depth (in m) pairs. The format is repeated for each of the radials. Next, the process is repeated for sensor #2, and so on, as defined in Table 7 and illustrated in Figure 14

Table 7. NUWbathy.inp file structure.

Line #	*Entry*	*Notes*
1	Number of radial bearings, nr	Number applies to all sensors and the transmitter
2	Radial bearing; number of bathymetry points n; target radial flag	For sensor #1. Degrees measured from line between sensor and transmitter; number of points currently limited to 500 per radial; flag identifying potential target radials
3 to (3+n-1)	Range; depth of bathymetry	Km; m (n range/depth pairs, one pair per line)
	Repeat line 2 to line 3 + n - 1 for each radial for sensor #1	
	Repeat line 2 to line 3 + n - 1 for each radial for sensor #2	
	Repeat line 2 to line 3 + n - 1 for each radial for sensor #nrd	
	Repeat line 2 to line 3 + n - 1 for each radial for transmitter	

```
4                        !number of radials- same for all assets
0. 11  0                 !1st radial for sensor #1:   phi(deg),  #pts,   0/1 for target bearing
  0.0   238              !range,  depth of bathy for given #pts
  4.6   256
 18.5   238
 22.2   219
 33.3   183
 36.1   165
 38.0   146
 40.7   128
 41.7   110
 51.9    91
 55.6    91
90. 7  0                 !2nd radial for sensor #1:   phi(deg),  #pts,   0/1 for target bearing
  0.0   238              !range,  depth of bathy for given #pts
 22.2   219
 24.1   183
 27.8   165
 31.5   146
 46.3   128
 55.5   146
180.  12  0              !3rd radial:   phi(deg),  #pts,   0/1 for target bearing
  0.0   238
  2.8   219
    . . .
                         !repeat for all sensors plus transmitter radials
```

Figure 14. Example of NUWbathy.inp file

3.1.3 NUWsvp.inp

This file contains the bearing and range-dependent sound speed profiles for each asset. Currently the dimensions of the sound speed arrays are limited to 100 profiles along each bearing, and each profile can have at most 200 points. Ensure the order of sensors matches that used in NUWBathy.inp. The number of recievers *nrd* is on line 1 in NUWSEinput.inp, so the number of assets is *nrd*+1 to include the transmitter. In addition, the number of radial bearings is identical for all assets, and this the number matches that indicated in line 1 of NUWbathy.inp. The values of the radials must match those specified in NUWbathy.inp. Table 8 defines the format of the file and Figure 15 shows an example file.

Table 8. NUWsvp.inp file structure.

Line #	***Entry***	***Notes***
1	Number of sound velocity profiles along first radial, for receiver #1 bearing	Limited to 100
2	Range to profile; number of svp points, n	Km; # pts limited to 200
3 to 3+n-1	Depth; speed or temperature	m; m/sec or degrees C
	Keep repeating from line 1 to line 3+n-1for each subsequent radial bearing, for receiver #1	
	Repeat from line 1 for each subsequent receiver, and the transmitter	

```
1                          ! number of svp on this bearing for the first receiver,
0.       18                ! 0deg bearing , number of points in svp
0.    1498.0               ! 18 lines of depth and sound speed
30.0   1499.2
35.0   1491.7
40.0   1483.8
45.0   1475.4
50.0   1466.5

    . . .
125.0  1479.6
150.0  1482.1
175.0  1484.6
200.0  1488.4
225.0  1489.5
250.0  1490.6
300.0  1490.5
1                 ! number of svp on this bearing for the first receiver,
90.      18       !90 deg bearing , number of points in svp
0.    1498.0
30.0   1499.2
35.0   1491.7
. . .                          Repeat for each bearing and asset
```

Figure 15. Example of portion of NUWsvp.inp file.

3.1.4 NUWbottomloss.inp

This file contains bottom loss information for each asset and bearing. The type of bottom selected (MGS, geoacoustic, or table) will apply to all assets and bearings. Currently the dimensions of the loss arrays are limited to 25 different regions along each bearing, and for table entries, the number of points is limited to 91. Ensure the sensors match the order used in NUWBathy.inp. The number of receivers *nrd* is on line 1 in NUWSEinput.inp, so the number of assets is *nrd*+1 to include the transmitter. Also there must be the same number of radial bearings for each asset., as specified on line 1 of file NUWbathy.inp. The format is defined in Table 9 and an example shown in Figure 16.

Table 9. NUWbottomloss.inp file structure.

Line #,	***Entry***	***Notes***
1	Bottom treatment option; attenuation units	'XY': 'X'= {M,A,T} 'Y'= {F,M,W,N}
2	Number of range dependent bottom sets, n	
3 to 3+n-1	If 'X'='M': range; province number If 'X'='A': range; c1; rho1; atten1; h1; c2; rho2; atten2 If 'X'='T': range; # of table rows; then loop over # table rows with angle; reflection coefficient; phase	Km; MGS province number Km; m/sec; g/cc; units of 'Y'; m; m/sec; g/cc; units of 'Y' Km; number of rows; Degrees; decimal fraction; degrees
	Repeat from line 2 for each subsequent radial bearing, for receiver #1	
	Repeat from line 2 for each subsequent asset	

In this table, the following are definitions apply:

'X' = the bottom treatment option
- 'M' = MGS or HFBL provinces
- 'A' = Geoacoustic fluid layers (no shear)
- 'T' = Read in table of pressure reflection coefficients and phases as a function of grazing angle

'Y' = The attenuation units which are used in the geoacoustic layers only, choices are:
- 'F' = dB/(m kHz)
- 'M' = dB/m
- 'W' = dB/wavelength
- 'N' = nepers/m

```
 'AF'                                  !A = Geoacoustic fluid layers, F=attenuation units: db/m kHz
 4                                     !4 bottom loss rows for 0 deg bearing  - receiver #1
 0.0  1453.  1.41   0.038  10.  1557.  1.73   0.156
26.8  1547.  1.72  0.158  20.  1880.  2.175  0.085
31.5  1630.  2.00  0.157    4.  1880.  2.175  0.085
40.7  1772.  2.11  0.142  10.  1880.  2.175  0.085
 4                                     !90 deg  receiver
 0.0  1453.  1.41    0.038  10.  1557.   1.73  0.156
24.0  1547.  1.72    0.158  20.  1880.   2.175 0.085
27.8  1807.  2.175  0.131  30.  3500.  2.60   0.020
46.3  1630.  2.00    0.157    4.  1880.  2.175  0.085
 6                                     !180 deg  receiver
 0.0   1453.  1.41   0.038  10.  1557.  1.73  0.156
14.8  1807.  2.175 0.131  30.  3500.  2.60  0.020
19.4  1453.  1.41   0.038  10.  1557.  1.73  0.156
33.3  1807.  2.175 0.131  30.  3500.  2.60  0.020
44.4  1630.  2.00   0.157   4.  1880.  2.175 0.085
50.0  1772.  2.11   0.142  10.  1880.  2.175 0.085
 6                                     !290 deg  receiver
 0.0  1453.  1.41  0.038  10.  1557.  1.73  0.156
       . . .
```

Figure 16. Example of portion of NUWbottomloss.inp showing geoacoustic parameters for several bearings.

3.1.5 NUWbeampat.inp

This file contains the beampatterns for sensors and transmitter. In the program, these beams will be assumed to be pointing along the target bearings, defined by the target bearing flag in NUWbathy.inp, line 1. Ensure sensors match the order used in NUWBathy.inp. The number of receivers, *nrd*, is specified on line 1 in NUWSEinput.inp. The number of assets is *nrd*+1 to include the transmitter. The format is defined in Table 10 and an example shown in Figure 17.

Table 10. NUWbeampat.inp file structure.

Line #	***Entry***	***Notes***
1	Number of vertical D/E angles in pattern, n	
2	Horizontal beamwidth; towed array flag	Degrees to 10 dB down points on either side of MRA. Flag = 0 or 1
3 to 3+n	D/E angle; loss	Degrees; dB
	Repeat from line 1 for next asset	

In this table, the towed array flag on line 2 refers to a value 0 (no) or 1 (yes) to indicate to the reverberation calculations whether to treat this array as a towed array broadside beam that will have a second beam with the same D/E pattern 180 degrees from the target bearing. There is presently no ability to specify towed array beams other than broadside; this is being added as a future enhancement.

```
3                    !number of sensor D/E angles
 1.8  1              !horizontal beamwidth of sensor (deg),    towed array flag: 0/1
-90.0   0.0          !angle(deg),   loss (dB)
0.0     0.0
90.0    0.0
37                   !number of transmitter D/E angles
 120.   0            !transmitter beamwidth (deg), towed array flag of 0/1
-90.0   51.4         !angle(deg), loss dB
-80.0   21.2
-70.0   28.5
-60.0   19.85
-50.0   20.55
-45.0   25.3
-40.0   19.0
-35.0   14.0
-30.0   13.66
-25.0   20.0
-20.0   18.44
-17.5   11.69
-15.0   7.72
-12.5   5.00
-10.0   3.05
-7.5    1.66
-5.0    0.721
-2.5    0.178
0.0     0.0
2.5     0.178
 . . .
```

Figure 17. Example of portion of NUWbeampat.inp.

3.2 Output files

3.2.1 Arrival.txt

This file contains incoherent ray arrival structures. As illustrated in Figure 18, the file begins by listing the frequency and source depth, then the bearing angle and sensor number. Next it lists by column the target depth(m), range(km), acoustic intensity, phase(rad), delay time(sec), source angle(deg), receiver angle(deg), number of reflections from the surface, and the number of reflections from the bottom. A header with abbreviations of these outputs is given for the reader's convenience. The Fortran output format for these numbers is (f7.1,f7.2,2e12.4,f7.3,2f7.2,2i4). This listing is repeated for each bearing that was designated a target bearing in the input file NUWbathy.inp and for each sensor and the transmitter (last listings). The write statements are in subroutine Writearrival in the file writeoutput_active.f90.

The tables include all the target depths that were specified in NUWSEinput.inp on line 8. They also include entries for the surface and bottom to enable reverberation to be computed. The surface entries are listed first at 0 depth, then come the target depth entries, followed by the bottom entries. Note that the depths listed for the bottom entries change with range as the bottom contour changes.

The amplitude in this table is the incoherent acoustic intensity. The transmission loss from these entries is TL=10*log(sum of entries at the same range and depth). It is possible to change the default run choice setting to produce a coherent SALT table, however in that case, the amplitude in the table is an acoustic pressure and the phase and delay time must be used to produce a coherent intensity. Since the other active products of reverberation, signal time series and signal excess are programmed to use intensity inputs, the change to a coherent calculation would not provide the correct input. A plot of the information provided in an arrival.txt file is shown in Figure 19.

```
 1200Hz  18.30m source depth
290. deg Bearing  1 sensor number
Tdepth  Range      Amp   Phase-rad   Time  Sangle Rangle Ntop Nbot
   0.0   0.76  0.4199E-12  0.5960E+01  0.609 -32.38  32.74   2   1
   0.0   0.76  0.1078E-10  0.2091E+00  0.609 -32.26  32.64   2   1
   0.0   0.76  0.1423E-09  0.6779E+00  0.609 -32.14  32.52   2   1
   0.0   0.76  0.7818E-09  0.1023E+01  0.609 -32.03  32.40   2   1
   0.0   0.76  0.2647E-08  0.1275E+01  0.609 -31.79 -32.17   1   1
   0.0   0.76  0.2856E-09  0.1633E+01  0.609 -31.56 -31.93   1   1
   0.0   0.76  0.1404E-05  0.0000E+00  0.506  -0.35  -1.51   1   0
```

Figure 18. Portion of output file arrival.txt showing some of the surface entries.

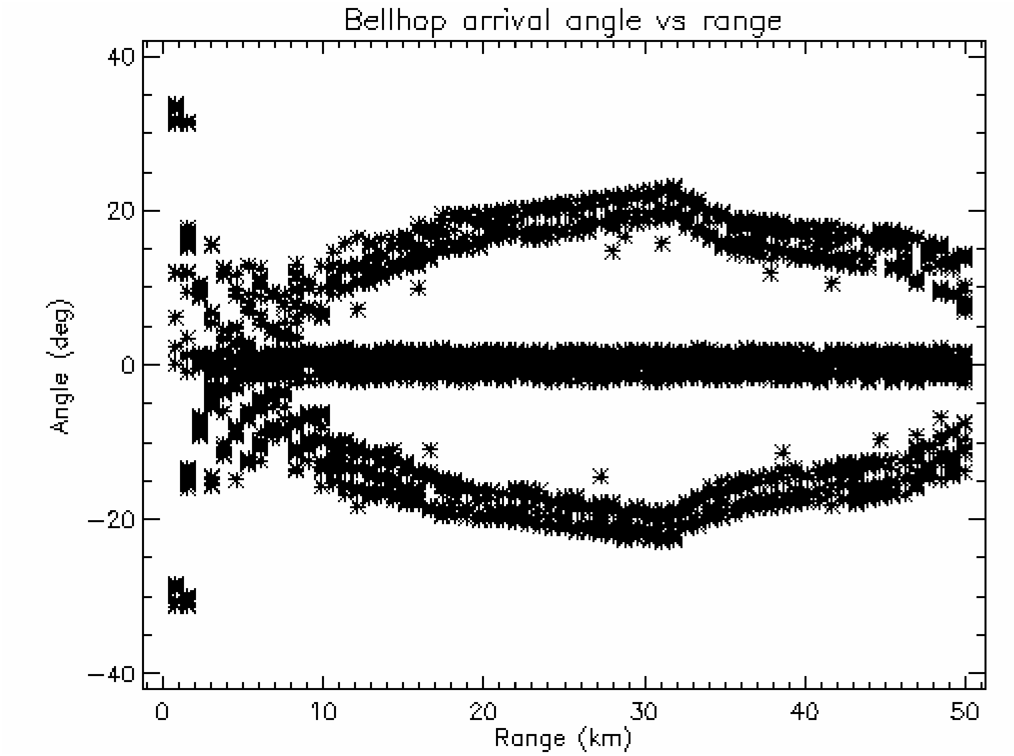


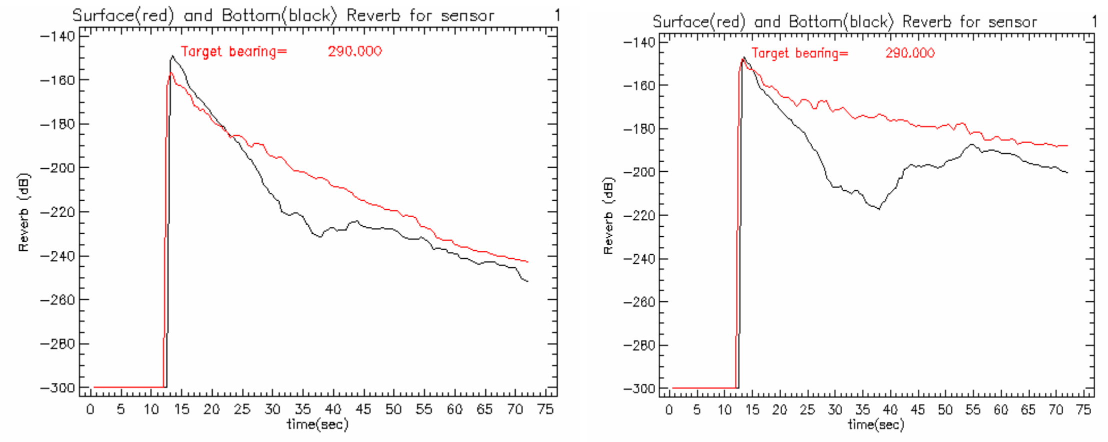
Figure 19. Example of arrival angle vs range plotted using arrival.txt for the 21m target depth.

3.2.2 Reverb.txt

This file contains the surface and bottom reverberation as a function of time, with the source level removed. As illustrated in Figure 20, the first line contains the number of time points, the sensor number and depth, the transmitter depth and the target bearing as specified in NUWbathy.inp. Next is a listing of the time array. Then the bottom reverberation time series in dB (without SL), and the surface reverberation time series in dB (without SL). This file structure is repeated for each target bearing and each receiving sensor. The write statements are in subroutine Writerevb in the file writeoutput_active.f90.

```
    144          1   18.3     21.00   290.0000
  0.5000000     1.000000     1.500000     2.000000     2.500000
   3.000000     3.500000     4.000000     4.500000     5.000000
   5.500000     6.000000     6.500000     7.000000     7.500000
      . . .
-300.0000    -300.0000    -300.0000    -300.0000    -300.0000
  -151.3029    -148.7673    -151.4195    -153.2674    -154.5864
  -157.0401    -162.3014    -164.1788    -165.6530    -168.0730
  -168.6941    -170.1840    -172.5353    -173.5714    -175.9944
     . . .
```

Figure 20. Selected portions of Reverb.txt output file.



(a) (b)

Figure 21. Plots of surface and bottom reverberation from reverb.txt (note: source level is not applied).

The -300 dB value shown at the start of the reverberation section of the listing is a default value designating no reception at that time.

While the reverberation is an important computation in its own right, this file is primarily intended to be combined with the signal.txt file to produce the signal excess that may be computed at a later date or in another language. The source level is not included to make the signal excess computation more flexible.

In Figure 21a, the plot illustrates the use of the Chapman Harris (surface) and Lambert (bottom) scattering strengths and the Beckmann Spezzichino surface loss as listed in NUWSEinput.inp. By way of contrast, the Figure 21b used Ogden Erskine (surface) and Ellis Crowe (bottom) scattering strengths and Modified Eckart surface loss. This latter choice of models produces a much larger contribution from the Sambro bank (shown in Figure 9).

3.2.3 Signal.txt

This file contains the signal time series for each receiver's range, target depth and target bearing. The source level and target strength are not included in this output. At present there are no graphics that make use of this file's output. It is provided to be an input along with the reverb.txt file for signal excess calculations that may wish to be computed at a later date or in another language. The source level and target strength are not applied to this file to make the signal excess computation more flexible. The write statements are in subroutine Writesignal in the file writeoutput_active.f90.

3.2.4 SE.txt

This file contains the signal excess computed using the Fortran file SE_active.f90, which is included in this code package. The SE computation begins by working in intensity units and multiplying the signal time series by the source level and target strength. It then adds the surface and bottom reverberation intensities, multiplies by the source level and adds the noise intensity to form the interference level power sum. Taking the maximum signal to interference ratio from the time series, the program converts this to dB and subtracts the dB values of detection threshold and system loss to form the signal excess. This result is smoothed over range using a Savitsky-Golay smoothing filter found in subroutine smoother.

The SE is listed in the file SE.txt. The first line contains the frequency and transmitter depth. The next line contains the number of ranges, the number of target depths, the number of target bearings (those that were identified in NUWbathy.inp), and the number of receiving sensors. The range array is listed next. Following this, the output loops over sensor number and writes the sensor number and sensor depth. Inside this sensor loop, the output loops over the target bearing and writes its value in degrees. Lastly, the output loops over target depth and writes each depth in metres, followed by the SE(dB) vs range array. The write statements are in subroutine WriteSE in the file writeoutput_active.f90. An example SE.txt file is shown in Figure 22 and a plot of signal excess against range in Figure 23.

```
Frequency=        1200  transmitter depth (m)=   21.00000
        66          2          1          1
  0.7575104      1.515021       2.272531       3.030042       3.787552
   4.545063       5.302573       6.060083       6.817594       7.575105
   8.332615       9.090125       9.847635       10.60515       11.36266
   12.12017       12.87768       13.63519       14.39270       15.15021
   15.90772       16.66523       17.42274       18.18025       18.93776
   19.69527       20.45278       21.21029       21.96780       22.72531
   23.48282       24.24033       24.99784       25.75536       26.51287
   27.27037       28.02789       28.78540       29.54291       30.30042
   31.05793       31.81544       32.57295       33.33046       34.08797
   34.84548       35.60299       36.36050       37.11801       37.87552
   38.63303       39.39054       40.14806       40.90556       41.66307
   42.42059       43.17809       43.93560       44.69312       45.45063
   46.20814       46.96565       47.72316       48.48067       49.23818
   49.99569
 1   18.30000       21.00000    sensor# and depth, xmitter depth
   290.0000    target bearing (deg)
   21.00000    target depth (m)
   9.256289       8.535771       9.359191       9.144906       8.026057
   7.084254       4.875857       3.116795       1.304128      0.3261878
 -0.1079665      -1.899835      -3.688573      -4.459661      -5.735830
  -7.032741      -8.430945      -9.686783      -11.37277      -12.41770
  -13.60939      -14.89992      -15.97495      -17.22614      -18.70700
  -19.57372      -21.09822      -22.91412      -24.28225      -25.03448
  -26.24449      -27.77931      -28.72955      -29.85480      -31.56085
  -32.87404      -34.36069      -35.25083      -36.51894      -37.97610
  -38.86609      -39.71757      -41.35246      -42.44070      -43.37093
  -44.78685      -46.59248      -47.37603      -48.62641      -50.29294
  -51.74666      -52.79663      -53.94806      -54.60282      -54.82505     [continued next page]
```

-55.44629	-56.15428	-56.84904	-57.50505	-58.19662
-68.52665	-58.72824	-60.37620	-61.14380	-63.53962
-273.1898				
80.00000	target depth (m)			
-18.53436	-20.66035	-23.45649	-25.00367	-28.71472
-31.48188	-35.14766	-38.54745	-41.59624	-44.67906
-47.18526	-50.00219	-52.73748	-54.50350	-56.60928
-59.08985	-62.29815	-65.48646	-67.66393	-69.63187
-71.12508	-73.26772	-76.85026	-79.01120	-80.86200
-82.96901	-84.39195	-85.44653	-84.95473	-84.61140
-83.78305	-83.09189	-82.90885	-82.79535	-82.34694
-82.36971	-82.73310	-83.66893	-83.49229	-83.22469
-82.87249	-83.55067	-84.22935	-84.87180	-85.25072
-85.72158	-85.85942	-86.47194	-86.91490	-87.88622
-88.32675	-88.77820	-89.07786	-89.83282	-90.16600
-90.51435	-90.68188	-91.16041	-91.51222	-91.77579
-100.5740	-92.59559	-92.36211	-93.96441	-95.86541
-277.0495				

Figure 22. Example of portion of SE.txt output.

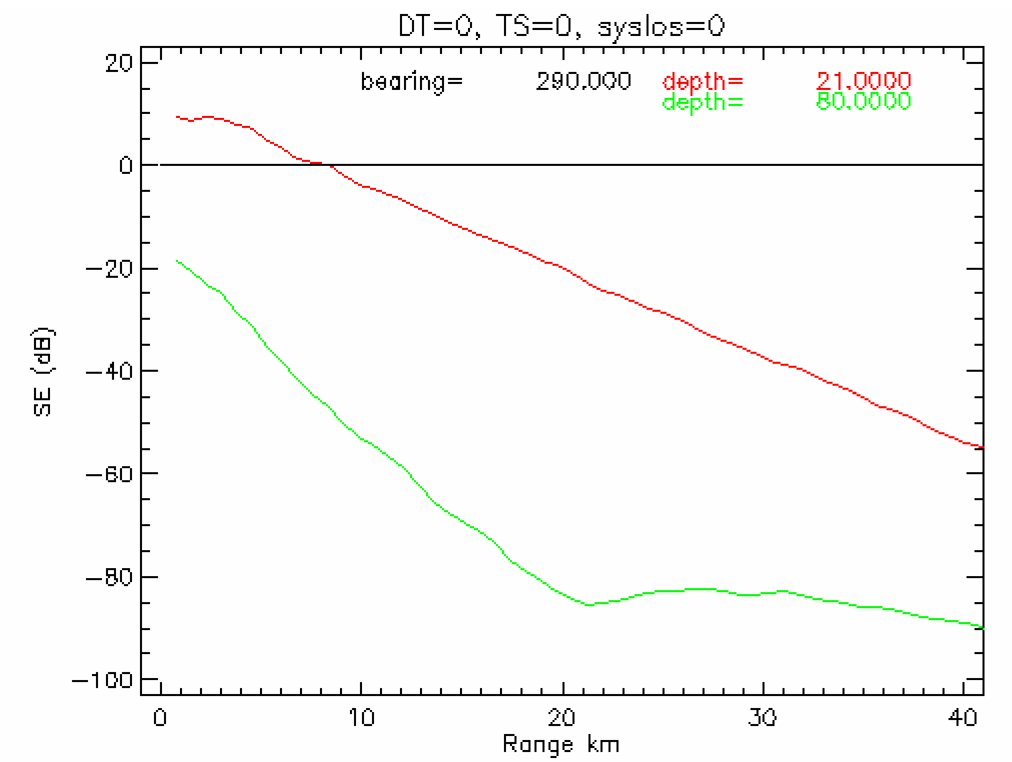


Figure 23. Example of signal excess plot, with DT=TS=syslos=0.

3.2.5 TL.txt

This file contains the transmission loss as a function of range for the transmitter location to the target depth, and for the sensor location to the target depth. The beam patterns of the transmitter and sensor are included in the calculation. The output file consists of a loop over sensor number and target bearing, with lines containing the number of ranges, the number of target depths, then the sensor number, sensor depth, transmitter depth and the target bearing. Following this is the range array. Then in a loop over target depth, the depth is listed, followed by the transmitter transmission loss array vs range, and lastly the sensor transmission loss array vs range. The write statements are in subroutine WriteTL in the file writeoutput_active.f90. An example TL.txt file is shown in Figure 24 and a plot of transmission loss against range in Figure 25.

```
        66          2
         1   18.30000       21.00000       290.0000
0.7575104      1.515021      2.272531      3.030042      3.787552
 4.545063      5.302573      6.060083      6.817594      7.575105
 8.332615      9.090125      9.847635      10.60515      11.36266
 12.12017      12.87768      13.63519      14.39270      15.15021
 15.90772      16.66523      17.42274      18.18025      18.93776
 19.69527      20.45278      21.21029      21.96780      22.72531
 23.48282      24.24033      24.99784      25.75536      26.51287
 27.27037      28.02789      28.78540      29.54291      30.30042
   ...
```

Figure 24. Example of portion of TL.txt output

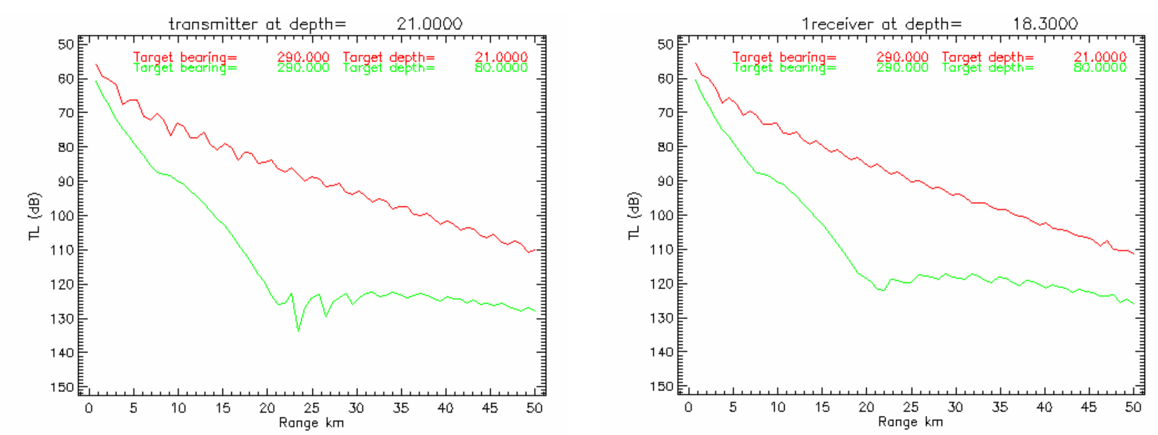


Figure 25. Example of TL plot from the TL.txt file. Left: transmission loss from transmitter to target, including transmitter beam pattern. Right: transmission loss from receiver to target, including receiver beam pattern. Two target depths are shown.

3.2.6 Bellhop_active.log

This file contains a log of the runtime statements generated in any run. Some inputs are echoed, and any warnings or errors are listed here as generated by the Bellhop code.

3.3 Plot Routines

Some IDL plot routines have been prepared to provide simple graphic representations of the active program outputs. These should be freely altered to suit the users' data and output requirements.

Read_arrivals_plot_angleortimes.pro – reads SALT table output in arrival.txt and plots arrival angle vs range as shown in Figure 19.

Read_reverb_plot_reverb.pro - reads reverb.txt and plots surface and bottom reverberation as shown in Figure 21.

Read_SE_plot_SE.pro – reads SE.txt and plots signal excess vs range as shown in Figure 23.

Read_TL_plot_TL_active.pro – reads TL.txt from the active program output and plots receiver to target transmission loss and transmitter to target transmission loss in two windows as a function of range as shown in Figure 25.

4. Boundary loss

To help understand the transmission loss predictions a separate program called boundaryloss is provided that will compute the bottom and surface losses. The file components for this algorithm are:

Boundaryloss.f90 – main program

- READIN_V3- reads runinput.inp
- READSVP_v3 – reads speed.inp
- READBOTLOSS_v3 – reads bottomloss.inp
- BOTT_NEW – function to compute MGS bottom loss from province numbers
- SURF_NEW – function to compute surface reflection coefficients using Bechmann-Spezzichino formulas
- LFSOPN – computes surface loss per bounce for open ocean using Modified Echart
- TMP_SSP – function to convert temperature to speed using Leroy's equation
- twolayerRefl – computes reflection coefficient from two fluid layers of sediment
- CRCIS – function to convert real wave speed and attenuation to a single complex wave speed
- ERROUT – outputs error messages

4.1 Input files

4.1.1 Runinput.inp

This is the same file that BellhopDRDC_ray_TL_v3 uses, described in section 2.1.1. The boundary loss algorithm uses the run title, frequency, wind speed and surface loss model from this file.

4.1.2 Speed.inp

This is the same file that BellhopDRDC_ray_TL_v3 uses, described in section 2.1.2. The boundary loss algorithm uses the sound speed from the surface and the bottom of the first profile in this file.

4.1.3 Bottomloss.inp

This is the same file that BellhopDRDC_ray_TL_v3 uses, described in section 2.1.3. The boundary loss algorithm uses all lines in this file.

4.2 Output files

4.2.1 Botloss.txt

This file contains the bottom loss and grazing angles for each bottom description in the input file. The structure of this file begins with the run title, the frequency, the number of range dependent bottom's being specified, and a string showing the type of bottom being computed, such as the two layer model, the MGS model or a user input table. Next the number of grazing angles is listed, followed by an array of grazing angles then an array of dB losses. These last three are repeated for each bottom specified. An example of this file is shown in Figure 26.

An example of the plot of these data is shown in Figure 27. The different curves are labelled according to their order in the bottomloss.inp file.

```
BELLHOP- Emerald basin toward Sambro Bank

  1200.000
         7
 two-layer fluid reflection bottom loss
        91
         0        1        2        3        4        5
         6        7        8        9       10       11
        12       13       14       15       16       17
        18       19       20       21       22       23
        24       25       26       27       28       29
        30       31       32       33       34       35
        36       37       38       39       40       41
        42       43       44       45       46       47
        48       49       50       51       52       53
        54       55       56       57       58       59
        60       61       62       63       64       65
        66       67       68       69       70       71
        72       73       74       75       76       77
        78       79       80       81       82       83
        84       85       86       87       88       89
        90
 5.8247424E-03  0.5140966   0.9780043   1.415306   1.952464
  2.857528   4.540282   6.287792   5.386145   3.989525
  3.439673   3.525668   3.814885   3.853908   3.888648
  4.145320   4.254133   4.699588   6.281219   8.995653
  10.13510   8.535233   8.418880   11.07827   15.56623  ...
```

Figure 26. Portion of a Botloss.txt file.

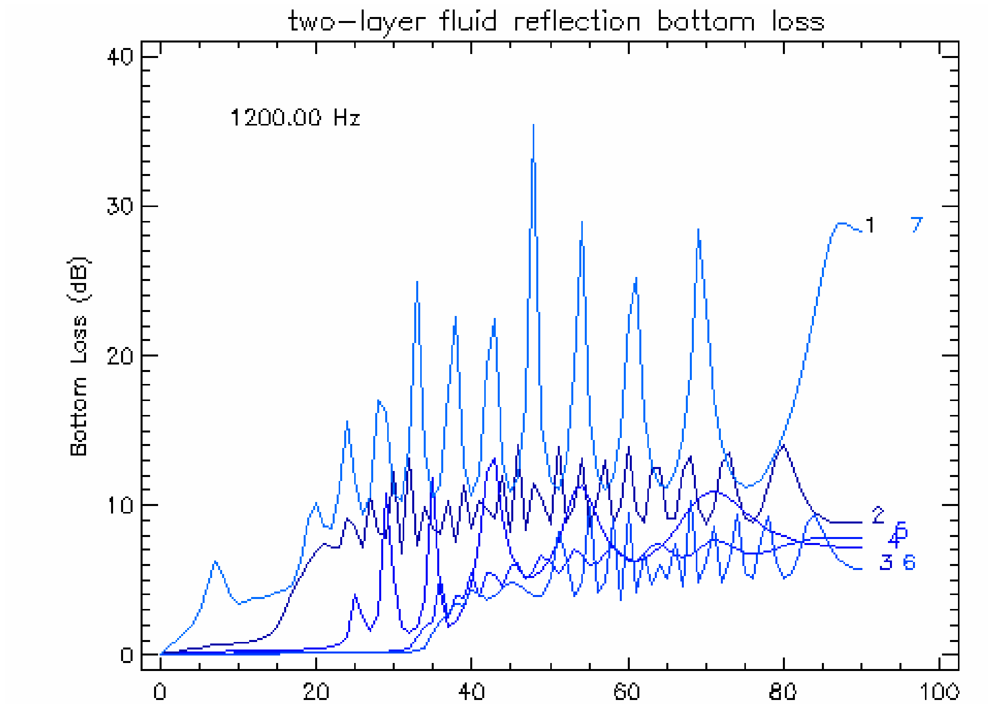


Figure 27. Plot of botloss.txt showing the seven regions listed in bottomloss.inp for the acoustic bottom descriptions using read_loss_plot_boundaryloss.pro.

4.2.2 Surfloss.txt

This file contains the surface loss and grazing angles from the selected loss algorithm. The structure of this file begins with the run title, the frequency, a string showing the surface loss algorithm, the number of points in the table, then an array of grazing angles and an array of losses in dB. An example of this file is shown in Figure 28. A plot of these data is shown in Figure 29 (where both surface loss models are displayed).

```
BELLHOP- Emerald basin toward Sambro Bank

 1200.000
 Beckman Spezzichino surface loss
        91
        0       1       2       3       4       5
        6       7       8       9      10      11
       12      13      14      15      16      17
       18      19      20      21      22      23
       24      25      26      27      28      29
       30      31      32      33      34      35
       36      37      38      39      40      41
       42      43      44      45      46      47
       48      49      50      51      52      53
       54      55      56      57      58      59
       60      61      62      63      64      65
       66      67      68      69      70      71
       72      73      74      75      76      77
       78      79      80      81      82      83
       84      85      86      87      88      89
       90
 0.0000000E+00  3.8063690E-02  7.6452129E-02  0.1151595      0.1541792
 0.2474124      0.3475395      0.4501680      0.5543828      0.6593952
 0.7645975      0.8695914      0.9741867      1.078379       1.182310
  1.286222       1.390416      1.495207       1.600906       1.707793
  1.816111       1.926067      2.037831       2.151542       2.267317  . . .
```

Figure 28. Example portion of surfloss.txt.

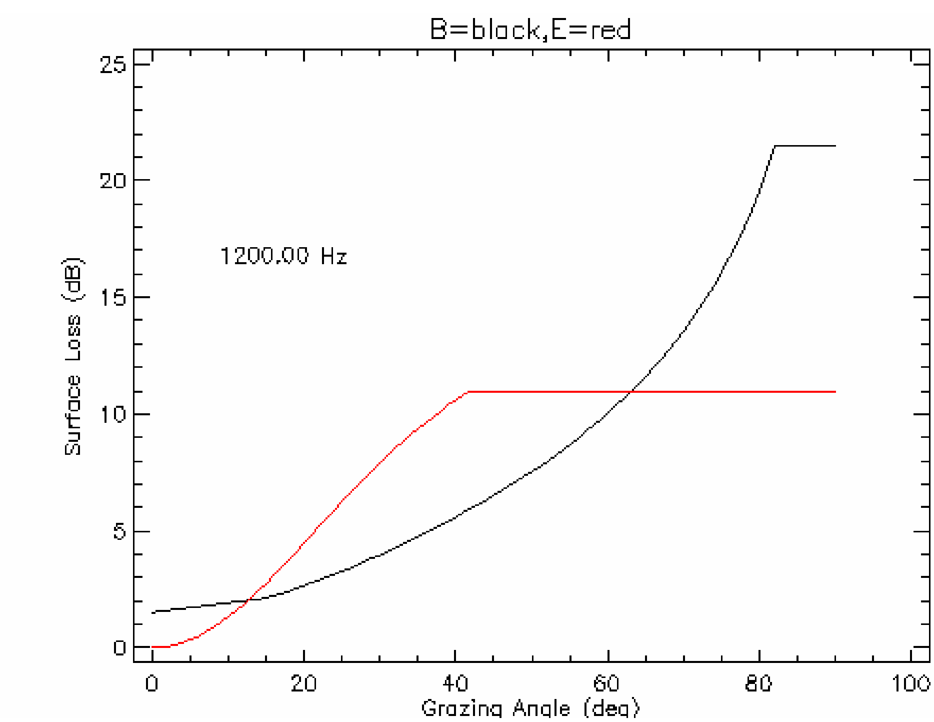


Figure 29. Plot of surfloss.txt for both Beckman Spezzichino (black) and modified Eckart (red).

4.3 Plot routine

An IDL plot routine has been prepared to provide a simple graphic representation of the boundaryloss outputs. This should be freely altered to suit the users' data and output requirements.

Read_loss_plot_boundaryloss.pro - Routine to read the output in Botloss.txt and Surfloss.txt and produce two loss figures.

List of symbols/abbreviations/acronyms/initialisms

CANTASS	Canadian Towed Array Sonar System
DND	Department of National Defence
DRDC	Defence R&D Canada
EMM	Environment Modeling Manager
GS	Grid search
HFBL	High frequency bottom loss
IDL	Interactive Data Language
MGS	Marine Geophysical Survey
NUW	Networked Underwater Warfare
R&D	Research & Development
Reverb	Reverberation
SALT table	Sound angle, level and delay time table
SE	Signal excess
SL	Source level
svp	Sound velocity profile
TDP	Technology Demonstration Project
TL	Transmission Loss

This page intentionally left blank.

Distribution list

Document No.: DRDC Atlantic CR 2006-067

LIST PART 1: Internal Distribution by Centre:

2	DRDC Atlantic Library File Copies
3	DRDC Atlantic Library (Spares)
1	Scientific Authority
1	LCdr Lex Stuart
1	Marcel Lefrançois
1	Jim Theriault
1	Garfield Mellema
1	John Osler
1	Paul Hines
1	Dale Ellis
1	Byron Topp
14	TOTAL LIST PART 1

LIST PART 2: External Distribution by DRDKIM

1	(CD) NDHQ/DRDKIM
1	McCammon Acoustical Consulting, 475 Baseline Road, Waterville NS B0P 1V0
1	LCdr Henrick Ouellet DMRS 5-3, HDHQ 101 Colonel By Drive Ottawa ON K1A 0K2
1	Laura Ozimek (DSTM and NUW TDP Project Director), DRDC, 305 Rideau St., Ottawa, ON K1A 0K2
2	SSO METOC MARLANT HQ PO Box 99000, Stn Forces Halifax, NS, B3K 5X5
6	TOTAL LIST PART 2

20 TOTAL COPIES REQUIRED

This page intentionally left blank.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

McCammon Acoustical Consulting

2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.)

UNCLASSIFIED

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C, R or U) in parentheses after the title.)

BellhopDRDC Users Guide, Version 3

Covering Passive and Active applications including Transmission Loss, Ray Tracing, Reverberation and Signal Excess

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Dr. Diana McCammon

5. DATE OF PUBLICATION (Month and year of publication of document.)

December 2006

6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)

38

6b. NO. OF REFS (Total cited in document.)

0

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Contract Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

DRDC Atlantic

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

11cg

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

W7707-05-3203

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)

10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

DRDC Atlantic CR 2006-067

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

(x) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

Over the most recent series of contracts the acoustic prediction model called Bellhop has been streamlined to more closely fit the requirements of DRDC Atlantic's Environment Modeling Manager. The Bellhop Fortran code has been streamlined by removing choices of interest in scientific research but less necessary in an operational system. The input data and file formats have been altered to satisfy the requirements of the controlling programs within the Environment Modeling Manager, and additional output capabilities for bistatic reverberation and active signal excess have been added. The program has been configured into two executables to run for passive or active prediction. This document provides a user's guide to the running of Bellhop, and describes some plotting routines available for viewing the prediction results.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Acoustic Prediction
Sonar
Modeling
Bellhop

This page intentionally left blank.

Defence R&D Canada

Canada's leader in defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca