



CAN UNCLASSIFIED



DRDC | RDDC
technologysciencetechnologie

AVAST Development

Michael Lichodzijewski
Lloyd's Register Applied Technology Group

Prepared by:
Applied Technology Group
Lloyd's Register Group
1888 Brunswick Street, Suite 400
Halifax, Nova Scotia
B3J 3J8 Canada
Contractor Document Number: TR-19-32, 14.28008.1152C
PSPC Contract Number: W7707-145679/001/HAL
Technical Authority: Layton Gilroy, Defence Scientist
Contractor's date of publication: March 2019

Defence Research and Development Canada

Contract Report
DRDC-RDDC-2019-C093
April 2019

CAN UNCLASSIFIED

CAN UNCLASSIFIED

IMPORTANT INFORMATIVE STATEMENTS

This document was reviewed for Controlled Goods by Defence Research and Development Canada using the Schedule to the *Defence Production Act*.

Disclaimer: This document is not published by the Editorial Office of Defence Research and Development Canada, an agency of the Department of National Defence of Canada but is to be catalogued in the Canadian Defence Information System (CANDIS), the national repository for Defence S&T documents. Her Majesty the Queen in Right of Canada (Department of National Defence) makes no representations or warranties, expressed or implied, of any kind whatsoever, and assumes no liability for the accuracy, reliability, completeness, currency or usefulness of any information, product, process or material included in this document. Nothing in this document should be interpreted as an endorsement for the specific use of any tool, technique or process examined in it. Any reliance on, or use of, any information, product, process or material included in this document is at the sole risk of the person so using it or relying on it. Canada does not assume any liability in respect of any damages or losses arising out of or in connection with the use of, or reliance on, any information, product, process or material included in this document.

- © Her Majesty the Queen in Right of Canada (Department of National Defence), 2019
- © Sa Majesté la Reine en droit du Canada (Ministère de la Défense nationale), 2019

CAN UNCLASSIFIED



Lloyd's
Register

Working together
For a safer world

AVAST Development

Technical Report # TR-19-32
Control Number: 14.28008.1152C

March 2019

A Lloyd's Register Technical Report

Prepared for:
Layton Gilroy
DRDC Atlantic
P.O. Box 1012
9 Grove Street
Dartmouth, Nova Scotia
Canada
B2Y 3Z7

Martec Limited is a member of the Lloyd's Register Group, doing business as the Applied Technology Group

Applied Technology Group tel. 902.425.5101
1888 Brunswick Street, Suite 400 fax. 902.421.1923
Halifax, Nova Scotia B3J 3J8 Canada email. LRATG-info@lr.org
www.lr.org/atg

Revision Control

Revision	Revision Date
00	27 March 2019

Proprietary Notice

This report was prepared under contract W7707-145679/001/HAL for the Defence Research and Development Canada – Atlantic Research Centre (DRDC-A).

The information contained herein may be used and/or further developed by DRDC-A for their purposes only.

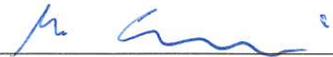
Complete use and disclosure limitations are contained in contract W7707-145679/001/HAL.

Signature Page

AVAST Development

TR-19-32

27 March 2019

Prepared by:  Date: 28 Mar 19
Michael Lichodzijewski
Lead Specialist – Software Development

Reviewed by:  Date: 28 Mar 19
Tom MacAdam
Software Development Team Leader

Approved by:  Date: 28 Mar 2019
David Whitehouse
Technical Manager

Executive Summary

The goal of this project was to further enhance the capabilities of the AVAST software.

The main application was extended to support multi-bistatic ray tracing analyses. The existing field point generation methods were consolidated, new field point generation and management methods were implemented, and all field point generation methods were made available for use with all AVAST solvers and sonar scenarios. Options for more realistic modelling of flat panel reflection were investigated. A new, more efficient process management algorithm was implemented in both the Windows and Linux versions of the software. The various software components were made more stable and updated to provide the user with better feedback.

Some areas for further study and improvement were also noted, with further study of flat panel reflection modelling and updates to the user interface identified as requiring particular attention.

Table of Contents

Revision Control	2
Proprietary Notice	3
Signature Page	4
Executive Summary	5
List of Figures	7
1. Introduction	8
2. Software Changes	9
2.1 Implementation of the Multi-Bistatic Ray Tracing Analysis	9
2.2 Improvements to the Field Point Generator	10
2.3 Improvements to the Modelling of Flat Panel Reflection	14
2.4 Improvements to Process Management	14
2.5 Improvements to Error Checking and Diagnostics	15
3. Conclusions and Recommendations	17
4. References	18
Appendix A	19
Appendix B	20
Appendix C	21
Appendix D	22
Appendix E	23

List of Figures

Figure 2-1: Button (bounded in red) used to launch a multi-bistatic analysis.....	9
Figure 2-2: Example of a results table produced for a multi-bistatic analysis.....	10
Figure 2-3: Dialog for defining field points along arcs.....	10
Figure 2-4: Dialog for defining field points along lines.....	11
Figure 2-5: Dialog for defining field points along X-Y plane (other planes similar).	11
Figure 2-6: Dialog for defining field points on a toroidal arc.	12
Figure 2-7: Dialog for manual entry and for copying and pasting of field points.	12
Figure 2-8: Field point data in a text editor.	12
Figure 2-9: Field point data in Microsoft Excel.	13
Figure 2-10: Field point data file format.	13
Figure 2-11: Menus for managing field point data.	13
Figure 2-12: Differences in the process management algorithms	15
Figure 2-13: Sample log file for the RayTracerApp application	16

List of Tables

Table 2-1: Comparison of execution times between the original and current process management algorithms.....	15
--	----

1. Introduction

AVAST is a software tool that provides the means to calculate acoustic Target Echo Strength (TES) using solvers that implement the Boundary Integral Equation Method (BIEM), the Kirchhoff Method, and ray tracing.

The goal of this task was to enhance the software by implementing support for mutli-bistatic ray-tracing analyses, consolidating and extending the field point generation and management methods, investigating ways to improve the modelling of flat panel reflection, optimizing process management, and generally improving the stability and usability of the software.

Details of the enhancements are provided in the following sections.

2. Software Changes

A number of software changes were made in order to address the requirements listed in the statement of work [1].

2.1 Implementation of the Multi-Bistatic Ray Tracing Analysis

The aim of this task was to extend the software to allow for parallelized execution of multi-bistatic ray-tracing analyses in the same manner as monostatic and single bistatic ray-tracing analyses.

The software was modified to separate field point generation and management from the different sonar scenarios. The workflows for bistatic and monostatic analyses were harmonized and the new workflow for multi-bistatic analyses was implemented.

The only remaining difference between the various sonar scenarios is how the field point data is used. Specifically, the bistatic analysis uses the projector and receiver positions as defined on the Analysis Setup page of the AVAST Property Data dialog; the monostatic analysis uses projector/receiver positions as defined by the field points; and the multi-bistatic analysis uses the field points to derive all possible combinations of projector and receiver positions.

The multi-bistatic analysis workflow is initiated by clicking the Multi BiStatic button on the Analysis Setup page of the AVAST Property Data dialog as shown in Figure 2-1 below.

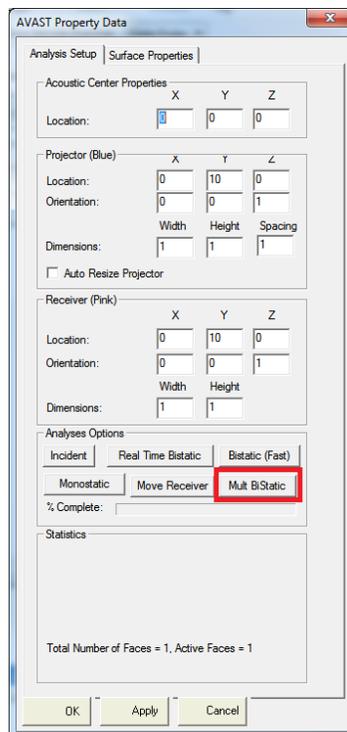


Figure 2-1: Button (bounded in red) used to launch a multi-bistatic analysis.

As with other types of analyses, the user is given the option of running the analysis locally, in either serial or parallel modes, or of exporting the data for external analysis. The user is also prompted to enter the name of a .mbs file to be used to store the results table similar to one shown in Figure 2-2. For locally-run parallel analyses and when exporting data for external analysis, the name of the user specified file is also used to form the names of input files that are generated for each analysis step (combination of projector and receiver positions) and for the index file that groups them together. For example, given a file called example.mbs and two field points, the software will produce four files. One file is generated for each combination of the projector and receiver positions, called example.mbs.1.input, example.mbs.2.input, example.mbs.3.input, and example.mbs.4.input, and an index file is generated with the name example.mbs.index. During the course of parallel analysis additional temporary files are generated, but they are subsequently removed by the software.

#	Ax	Av	As	Ex	Ey	Ez	Rx	Ry	Rz	R	Pr	Tr	Rx	Pa	Ra	Pd	Rd	Td
1	0	0	0	0	100	0	0	100	0	100	361201	31449	225	900	900	401.334	0.2225	7.43824
2	0	0	0	0	100	0	0	100	0	100	361201	31449	225	900	900	401.334	0.2225	7.43824
3	0	0	0	0	100	0	0	100	0	100	361201	31449	225	900	900	401.334	0.2225	7.43824
4	0	0	0	0	100	0	0	100	0	100	361201	31449	225	900	900	401.334	0.2225	7.43824

Figure 2-2: Example of a results table produced for a multi-bistatic analysis.

2.2 Improvements to the Field Point Generator

The goals of this task were to: 1) consolidate the existing field point generation capabilities, 2) add new features to allow the user to generate field points on toroidal arcs, 3) enter field points manually, 4) import and export field point data from and to text files, 5) copy and paste field point data from applications such as Microsoft Excel, and 6) make all of these field point definition and management features available to the BIEM, Kirchhoff, and ray-tracing solvers.

The existing field point generation capabilities that were identified and consolidated were for field points along arcs, lines, and axis aligned planes. The associated user interfaces are shown in Figures 2-3, 2-4, and 2-5 respectively.

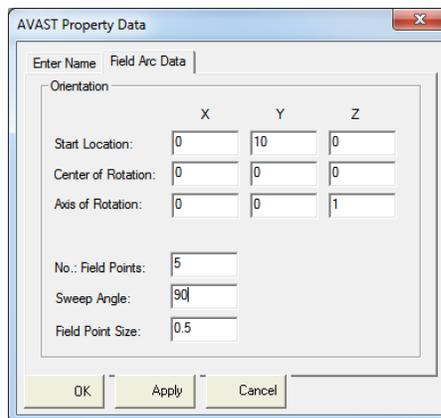


Figure 2-3: Dialog for defining field points along arcs.

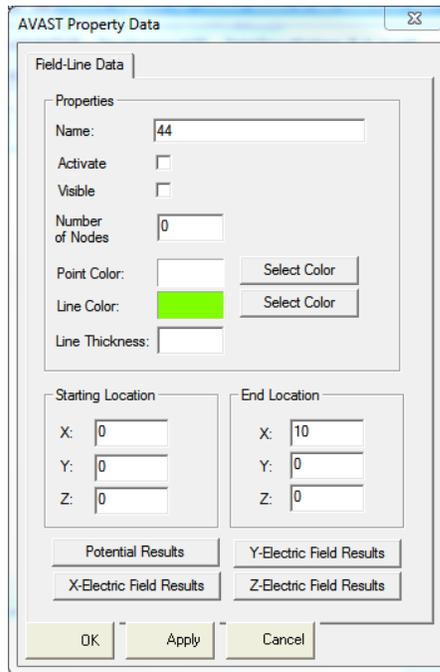


Figure 2-4: Dialog for defining field points along lines.

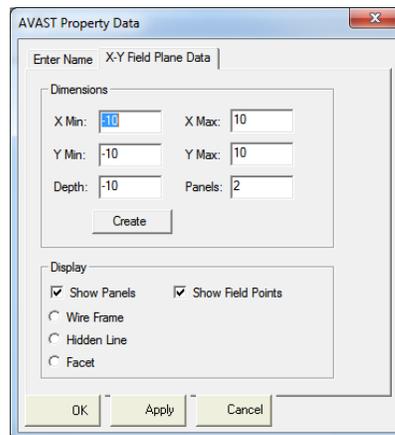


Figure 2-5: Dialog for defining field points along X-Y plane (other planes similar).

Several new capabilities were also implemented. The user interface for the newly implemented capability to define field points along toroidal arcs is shown in Figure 2-6.

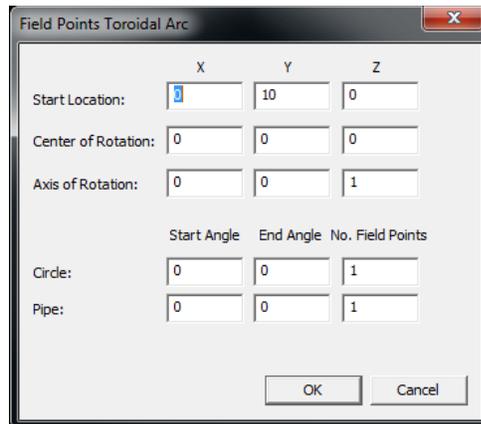


Figure 2-6: Dialog for defining field points on a toroidal arc.

Figure 2-7 shows the user interface for manual entry of field points and for copying and pasting field point data from text editors and Microsoft Excel as shown in Figure 2-8 and 2-9.

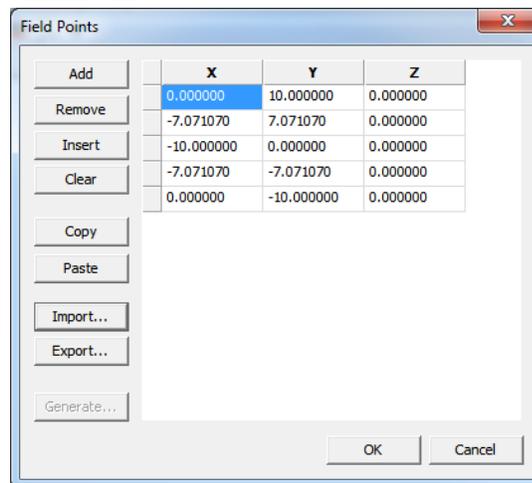


Figure 2-7: Dialog for manual entry and for copying and pasting of field points.

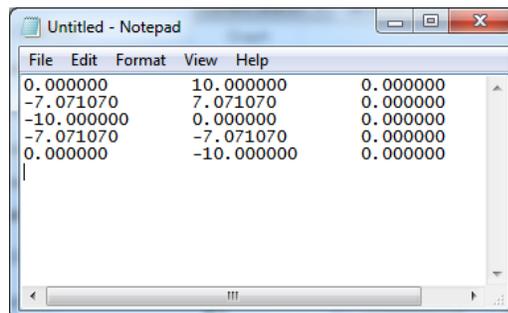
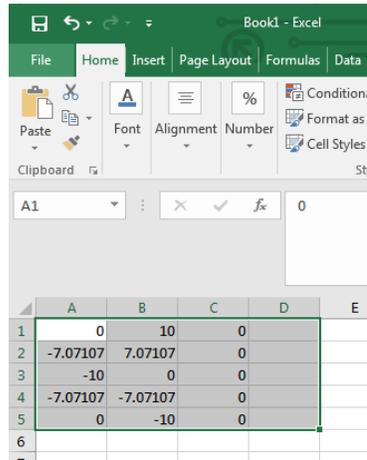


Figure 2-8: Field point data in a text editor.

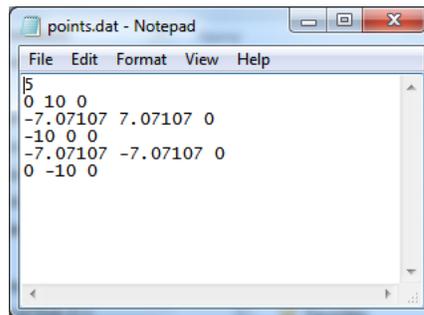


The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E
1	0	10	0		
2	-7.07107	7.07107	0		
3	-10	0	0		
4	-7.07107	-7.07107	0		
5	0	-10	0		
6					

Figure 2-9: Field point data in Microsoft Excel.

Consistent support for the import and export of field point data was also implemented using a file format that starts with a single line indicating the number of field points followed by the lines specifying the field point coordinates as shown in Figure 2-10.



```
points.dat - Notepad
File Edit Format View Help
5
0 10 0
-7.07107 7.07107 0
-10 0 0
-7.07107 -7.07107 0
0 -10 0
```

Figure 2-10: Field point data file format.

Pre-existing and newly implemented field point management features are exposed through the menu shown in Figure 2-11.

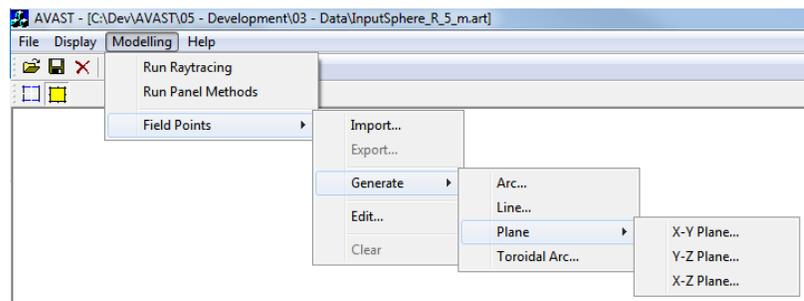


Figure 2-11: Menus for managing field point data.

All field point management menu options become available once a model is created or loaded from a file. Once created, the same field point definition can be used with the BIEM, Kirchhoff, and ray tracing methods.

2.3 Improvements to the Modelling of Flat Panel Reflection

The aim of this task was to investigate options for the modelling of flat panel reflection to address unrealistic results produced using the current implementation.

The current implementation calculates TES as:

$$\text{TES} = 10 * \log_{10}(d_r / d_p) + 20 * \log_{10}(R),$$

Where:

d_r is the density of the rays received, which is calculated as the sum of the intensities of the individual rays received, divided by the receiver's area,

d_p is the density of the rays projected, which is calculated as the sum of the intensities of the individual rays projected, divided by the area of the projector, and

R is the distance between the acoustic centre and the receiver.

Some potential options were discussed. The first involved the enhancement of the existing ray dispersion function to allow for geometrically equal distribution of variable ray densities. The second involved changes to the existing ray dispersion function to allow for constant ray intensities with a geometrically variable distribution of rays. However, these two options were deemed as unlikely to address the unrealistic results issue.

Another potential solution involved tracking rays reflected off flat panels and applying a correction factor.

Due to lack of time and budget none of these options were pursued further.

2.4 Improvements to Process Management

In work recently completed for Task 50 of this contract [2], the Target Echo Strength (TES) ray tracing algorithm was ported from Windows to Linux allowing it to run on DRDC-A's High Performance Computing (HPC) platform. The high level of parallelization afforded by this platform revealed some deficiencies in the management of processes resulting in the locking of resources and less than optimal run times.

In the original process management approach, a set of projector/receiver position pairs was considered a job. Each individual projector/receiver position pair was considered a task to be distribute amongst a user specified number of subtasks, with each subtask assigned to a process. The original approach processed one projector/receiver position at a time. The high likelihood of different computation times for each subtask (process) meant that completed subtasks (processes) would not terminate until the completion of the longest one. This effectively meant that processing could not move to the next projector/receiver position until this happened.

The new approach takes a more holistic view of the work. Effectively, it creates all subtasks before any processing starts. The algorithm makes use of the user specified number of processors. Each subtask (process) is monitored and when it completes, it is allowed to fully terminate, and the next subtask (process) is started. This results in theoretical runtime improvement of almost 50%.

The difference between the allocation of processing resources is shown in Figure 2-12, illustrating a difference in the processing time for a job consisting of two tasks broken up into 10 subtasks that are distributed amongst 5 processors.



Figure 2-12: Differences in the process management algorithms

Table 2-1 shows a comparison of actual execution times for a contrived example of a sphere of a 5 m radius with a projector and receiver 30 meters by 30 metres using a ray spacing of 0.05 metres and for a job consisting of four tasks executed on 5 processors.

Table 2-1: Comparison of execution times between the original and current process management algorithms.

Windows				Linux			
	Before	After	%		Before	After	%
1	38.9558	20.1127	52%	1	75.6541	38.93717	51%
2	38.6649	22.2648	58%	2	76.6448	40.65706	53%
3	37.6843	20.49438	54%	3	77.6896	47.03776	61%
4	37.3146	20.65007	55%	4	84.9936	41.05576	48%
5	37.9771	20.55542	54%	5	81.1207	40.79556	50%
Min	37.3146	20.1127	54%	Min	75.6541	38.93717	51%
Max	38.9558	22.2648	57%	Max	84.9936	47.03776	55%
Avg	38.11934	20.81547	55%	Avg	79.22056	41.69666	53%
Worst	37.3146	22.2648	60%	Worst	75.6541	47.03776	62%
Best	38.9558	20.1127	52%	Best	84.9936	38.93717	46%

2.5 Improvements to Error Checking and Diagnostics

The aim of this task was to improve error checking and diagnostics to prevent improper operation of the software and to provide better feedback regarding problems with the operation of the software.

Additional checks were added to validate command line parameters and check for the presence of the required input data. Warning messages were added to the RayTracerDriverApp2 and RayTracerDriver applications to alert the user of problems. Logging was added to facilitate diagnostics.

Figure 2-13 shows a sample log file for the RayTracerDriver application.

```
03/19/19 18:15:18 UTC : Registered log file...
03/19/19 18:15:18 UTC : Command line: .\02_After\RayTracerApp.exe .\02_After\InputSphere_R_5_m.art.bg.4.input.5.input
03/19/19 18:15:18 UTC : Projecting 72240 rays
03/19/19 18:15:18 UTC : Ray density 72240.000000 rays/unit area
03/19/19 18:15:25 UTC : Projected rays 72240
03/19/19 18:15:25 UTC : Received rays 225
03/19/19 18:15:25 UTC : Projected density 80.266667
03/19/19 18:15:25 UTC : Received density 0.222500
03/19/19 18:15:25 UTC : Target hits 22600
03/19/19 18:15:25 UTC : Generation 1 rays 72240
03/19/19 18:15:25 UTC : Generation 2 rays 45200
03/19/19 18:15:25 UTC : Generation 3 rays 45158
03/19/19 18:15:25 UTC : Generation 4 rays 22498
03/19/19 18:15:25 UTC : Density ratio 0.002772
03/19/19 18:15:25 UTC : Target strength 14.427948 dB
03/19/19 18:15:25 UTC : Average time per ray = 0.000131 seconds
03/19/19 18:15:25 UTC : Time 9.484261 seconds
03/19/19 18:15:25 UTC : Exiting RayTracerApp.exe
```

Figure 2-13: Sample log file for the RayTracerApp application

3. Conclusions and Recommendations

The latest set of changes to the AVAST software have further enhanced its capabilities and performance, laying a good foundation for future improvements.

Recommended future software development work includes further improvements to the analysis workflows, particularly changes to data file formats aimed at reducing their complexity and duplication of data, improvements to configurability, additional research into improving performance, a more extensive investigation of ways to improve the modeling of flat panel reflection, simplification and modernization of the software's user interface, and modernization of the solvers implementing the BIEM and Kirchhoff methods. Further study aimed at assessing the sensitivity of the ray tracing approach to factors such as projector and receiver dimensions, ray density, transparency and the number of reflected rays is also recommended.

4. References

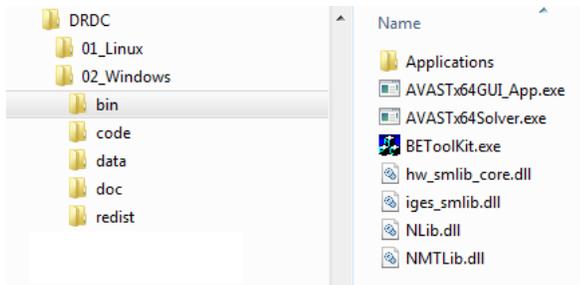
- [1] Contract W7707-145679/001/HAL TA-52 Statement of Work.
- [2] Lichodzijewski M. 2017. AVAST Ray Tracer for Linux. TR-17-12. Applied Technology Group. Halifax, Nova Scotia.

Appendix A

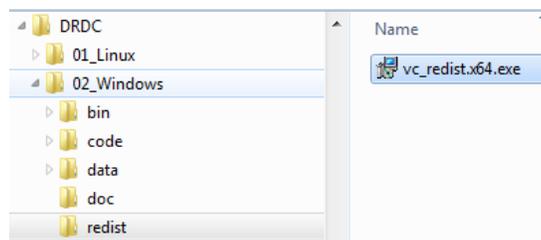
Installation Instructions for Microsoft Windows

To install the AVAST software on a Microsoft Windows system:

1. Go to the folder containing the provided AVAST application bundle
2. Go to the folder 02_Windows



3. Copy the contents of the bin folder to a folder of your choice.
4. Go to the folder 02_Windows\redistrib and run vc_redist.x64.exe



5. Test the application runs by launching BEToolKit.exe

Appendix B

Installation Instructions for Linux (CentOS 7)

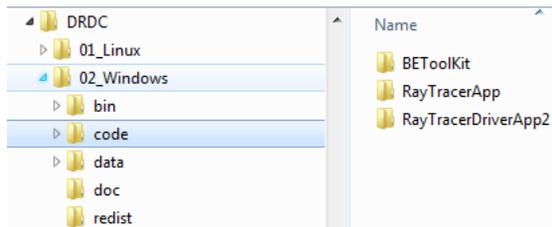
To install the AVAST software on Linux (CentOS 7), copy the files `RayTracerApp` and `RayTracerDriverApp2` found in the folder `01_Linux\bin` of the provided application bundle to a location of your choice. When complete, change the files' permissions to allow them to execute with `chmod +x`.

Appendix C

Building Instructions for Microsoft Windows

In order to build AVAST on Microsoft Windows do the following:

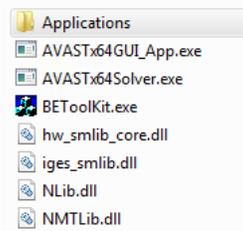
1. Copy the contents of the folder 02_Windows\code to a folder of your choice



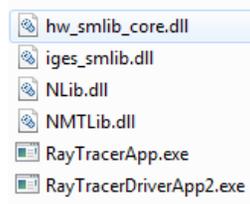
2. Place a copy of the folder containing the libraries and source code for SMLib 8.4.0 beside the folders for the AVAST applications.



3. Build one or more of the configurations of the solution BEToolKit.sln found in the folder BEToolkit
4. Build one or more of the configurations of the solution RayTracerApp.sln found in the folder RayTracerApp
5. Build one or more of the configurations of the solution RayTracerDriverApp.sln found in the folder RayTracerDriverApp2
6. In order to call the applications RayTracerApp.exe and RayTracerDriverApp2.exe from BEToolKit.exe, copy the files RayTracerApp.exe, hw_smlib_core.dll, iges_smlib.dll, NLib.dll, NMTLib.dll, and RayTracerDriverApp2.exe corresponding to a given configuration and place them in a folder called Applications that is in the same folder as BEToolKit.exe.



7. Ensure that the Applications folder has the following content:



Appendix D

Building Instructions for Linux

In order to build AVAST applications RayTracerApp and RayTracerDriverApp2 on Linux (CentOS 7) do the following:

1. Copy the contents of the folder 01_Linux to a location of your choice.
2. Ensure that a C++ compliant compiler is available. The version available with Devtoolset-7 is recommended.
3. To build the application RayTracerApp:
 - a. Ensure that the folder RayTracerApp and the folder SMLib8.4.0 are side by side.
 - b. Enter the folder RayTracerApp
 - c. Remove the file RayTracerApp, if it exists
 - d. Ensure that the folder objects exists and create it, if it does not
 - e. Remove the contents of the folder objects, if any.
 - f. Run the command `make -f RayTracerApp.mak`
 - g. Check for the presence of the executable file RayTracerApp
4. To build the application RayTracerDriverApp2:
 - a. Enter the folder RayTracerDriverApp2
 - b. Remove the file RayTracerDriverApp2, if it exists
 - c. Ensure that the folder objects exists and create it, if it does not
 - d. Remove the contents of the folder objects, if any
 - e. Run the command `make -f RayTracerDriverApp2.mak`
 - f. Check for the presence of the executable file RayTracerDriverApp2

Appendix E

Basic Operating Instructions

1. Use the Windows version of AVAST to model the scenario to be analysed
 2. Click the Bistatic (Fast), Monostatic, or Multi BiStatic button on the Analysis Setup page of the AVAST Property Data dialog to launch one of the analyses types
 3. Decide whether to run the locally or to export data for external execution
 4. For locally run analyses decide whether to run the analyses serially (using legacy code) or in parallel
 5. For locally run analyses specify the name of the file to store the results, run the analysis, and upon completion, open the previously specified results file
 6. For exported data specify the path for the data and the prefix to be used for the files for each step of the analysis (projector/receiver position) and for the index file. Creating a new folder for each set of exported data is recommended.
 7. Copy the exported data files to where they are to be used.
 8. On Windows run the command `RayTracerDriverApp2.exe prefix.index prefix RayTracerApp.exe 0`
 9. On Linux run the command `.\RayTracerDriverApp prefix.index .\RayTracerApp 0`
- The locations of the executables and the data files may need to be specified explicitly if not located in the same folder

Contact:
Michael Lichodzijewski
Lead Specialist – Software Development, Software Development

Applied Technology Group
1888 Brunswick Street, Suite 400
Halifax, NS
B3J 3J8

T 1-902-407-9819
F 1-902-421-1923
E michael.lichodzijewski@lr.org
W

www.lr.org/atg

March 2019

Lloyd's Register Group Limited, its subsidiaries and affiliates and their respective officers, employees or agents are, individually and collectively, referred to in this clause as 'Lloyd's Register'. Lloyd's Register assumes no responsibility and shall not be liable to any person for any loss, damage or expense caused by reliance on the information or advice in this document or howsoever provided, unless that person has signed a contract with the relevant Lloyd's Register entity for the provision of this information or advice and in that case any responsibility or liability is exclusively on the terms and conditions set out in that contract.

Lloyd's Register and variants of it are trading names of Lloyd's Register Group Limited, its subsidiaries and affiliates. Copyright © Martec Limited. "[Type year]". Martec Limited is registered under the Corporations Registration Act of Nova Scotia, Canada, doing business as Lloyd's Register Applied Technology Group (ATG). A member of the Lloyd's Register group.

DOCUMENT CONTROL DATA

*Security markings for the title, authors, abstract and keywords must be entered when the document is sensitive

1. ORIGINATOR (Name and address of the organization preparing the document. A DRDC Centre sponsoring a contractor's report, or tasking agency, is entered in Section 8.) Applied Technology Group Lloyd's Register Group 1888 Brunswick Street, Suite 400 Halifax, Nova Scotia B3J 3J8 Canada		2a. SECURITY MARKING (Overall security marking of the document including special supplemental markings if applicable.) CAN UNCLASSIFIED
		2b. CONTROLLED GOODS NON-CONTROLLED GOODS DMC A
3. TITLE (The document title and sub-title as indicated on the title page.) AVAST Development		
4. AUTHORS (Last name, followed by initials – ranks, titles, etc., not to be used) Lichodzijewski, M.		
5. DATE OF PUBLICATION (Month and year of publication of document.) March 2019	6a. NO. OF PAGES (Total pages, including Annexes, excluding DCD, covering and verso pages.) 25	6b. NO. OF REFS (Total references cited.) 2
7. DOCUMENT CATEGORY (e.g., Scientific Report, Contract Report, Scientific Letter.) Contract Report		
8. SPONSORING CENTRE (The name and address of the department project office or laboratory sponsoring the research and development.) DRDC – Atlantic Research Centre Defence Research and Development Canada 9 Grove Street P.O. Box 1012 Dartmouth, Nova Scotia B2Y 3Z7 Canada		
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.) 01ec - Naval Signature Management	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.) W7707-145679/001/HAL	
10a. DRDC PUBLICATION NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.) DRDC-RDDC-2019-C093	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.) TR-19-32; 14.28008.1152C	
11a. FUTURE DISTRIBUTION WITHIN CANADA (Approval for further dissemination of the document. Security classification must also be considered.) Public release		
11b. FUTURE DISTRIBUTION OUTSIDE CANADA (Approval for further dissemination of the document. Security classification must also be considered.)		

12. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Use semi-colon as a delimiter.)

Target echo strength; Acoustics; Ray tracing; Boundary Element Method; Kirchhoff

13. ABSTRACT/RÉSUMÉ (When available in the document, the French version of the abstract must be included here.)

The goal of this project was to further enhance the capabilities of the AVAST software. The main application was extended to support multi-bistatic ray tracing analyses. The existing field point generation methods were consolidated, new field point generation and management methods were implemented, and all field point generation methods were made available for use with all AVAST solvers and sonar scenarios. Options for more realistic modelling of flat panel reflection were investigated. A new, more efficient process management algorithm was implemented in both the Windows and Linux versions of the software. The various software components were made more stable and updated to provide the user with better feedback. Some areas for further study and improvement were also noted, with further study of flat panel reflection modelling and updates to the user interface identified as requiring particular attention.

Ce projet visait à améliorer davantage les capacités du logiciel AVAST. L'application principale a été étendue pour prendre en charge les analyses de traçage de rayons multi-bistatiques. Les méthodes de génération de points de terrain existantes ont été consolidées, de nouvelles méthodes de génération et de gestion de points de terrain ont été mises en œuvre et toutes les méthodes de génération de points de terrain ont été rendues disponibles pour une utilisation avec tous les solveurs AVAST et scénarios de sonar. Des options pour une modélisation plus réaliste de la réflexion sur écran plat ont été étudiées. Un nouvel algorithme de gestion des processus, plus efficace, a été mis en œuvre dans les versions Windows et Linux du logiciel. Les différents composants logiciels ont été rendus plus stables et mis à jour pour fournir à l'utilisateur un meilleur retour d'informations. Certains domaines nécessitant une étude et une amélioration plus poussées ont également été notés, avec une étude plus approfondie de la modélisation de la réflexion par écran plat et des mises à jour de l'interface utilisateur identifiées comme nécessitant une attention particulière.