AGORA v1.0

Airgun Array Signature Model

USER GUIDE

(Draft Version)

agora.model@gmail.com

H. Özkan Sertlek

University of Leiden, Institute of Biology, The Netherlands

Contents

1. INTRODUCTION 3
2. RUNNING AGORA 3
3. AGORA GRAPHICAL USER INTERFACE (GUI) 4
3.1. Simulation panel 4
3.2. Plotting panel - Frequency domain 6
3.3. Plotting panel - Time domain 8
4. AGORA SINGLE GUN 8
5. SUMMARY AND DISCUSSION 9
REFERENCES 10
ACKNOWLEDGEMENTS 10

1. INTRODUCTION

AGORA is a research algorithm which calculates the airgun array signatures to use in the sound mapping tools for the environmental risk assessment. The motivation of this algorithm is based on a PhD project (at Institute of Biology, University of Leiden) which aims to model the soundscape of the Dutch North Sea. This project is funded by ZKO programme of NWO (The Netherlands Organisation for Scientific Research).

In this document, the use of MATLAB GUI version of AGORA is described. The mathematical basement of the algorithm is described by [Sertlek and Ainslie, 2015].

Please copy the environment input files in the same directory with AGORA files. AGORA's Matlab source codes are stored in *"…\AGORA\AGORA_Matlab\"* folder. The executable file in "…\AGORA\AGORA_Windows_bin\" can be preffered (doesn't require Matlab, however, it requires the Matlab Runtime which can be downloaded http://nl.mathworks.com/products/compiler/mcr/). This version of AGORA is developed and tested for Windows operating systems. It is recommended to use 1280x1024 resolution.

2. RUNNING AGORA

"*AGORA_main.m*" loads the main GUI of AGORA. The default input parameters are stored in "*environment_default.ain*" file as

Table 1. The default input parameters in "environment_default.ain" file

Line number	***Quantity***	***Default Value***
Line1	*Duration[sec]*	*0.5*
Line2	*Temperature [degree]*	*22.2*
Line3	*Density [kg/m3]*	*1024*
Line4	*Sound speed [m/s]*	*1500.0*
Line5	*Tao*	*0.09115*
Line6	*Beta*	*0.520*
Line7	*Eta*	*0.8317*
Line8	*K*	*22230*
Line9	*Time step [s]*	*1e-4*

In order to calculate the time domain airgun signatures, *"agora_kernel"* function is used. This function iteratively solves the set of differential equations which is described in [*Sertlek and Ainslie,2015*]. After all airgun signatures are calculated, *"add_bubble_interactions"* function adds the bubble interactions as described by [Ziolowski,1982]. Next, the frequency domain signatures can be calculated by applying Fast Fourier Transform (FFT) as implemented in *"fft_func"* function. Finally, the energy dipole source level for each azimuth and elevation angle can be calculated by *"calc_directivity_AGORA_gui"* function based on [Duren,1988]. The time domain signature of airgun array haven't been implemented, yet. However, this can be calculated by applying inverse Fouirer transform to the output of *"calc_directivity_AGORA_gui"* function.

3. AGORA GRAPHICAL USER INTERFACE (GUI)

The user interface has been developed for AGORA to provide an user-friendly control of algorithm. If only interest is calculating the time domain signature of individual airguns, *"agora_kernel"* function can be directly used for your own algorithm without using AGORA GUI. It is described in Section 4.

AGORA GUI has three main panels as

- Simulation panel
- Plotting panel- Frequency domain
- Plotting panel- Time domain.

These panels are shown in Figure 1 and will be described in the next sections

3.1. Simulation panel

It is the panel where you can calculate or load the airgun array signatures. "*load signature*" button loads the pre-calculated airgun array signatures by AGORA. You should choose a precalculated signature data set from the list on the left of this button. These files are stored in the output folder of AGORA.

"simulator parameters" button open another panel (*Config Panel*) where you can modify and save the simulation parameters to *"environment.ain"* file. The default parameters are saved in a different file (*"environment_default.ain"*). The default parameters can be loaded from this panel. The default calibration coefficients are based on [Racca and Scrimger,1986] and calculated by an optimization approach in [MacGillivray,2006]. In order to increase the accuracy, these calibration coefficients can be re-calculated according to the recent measurement data based on the new airguns.

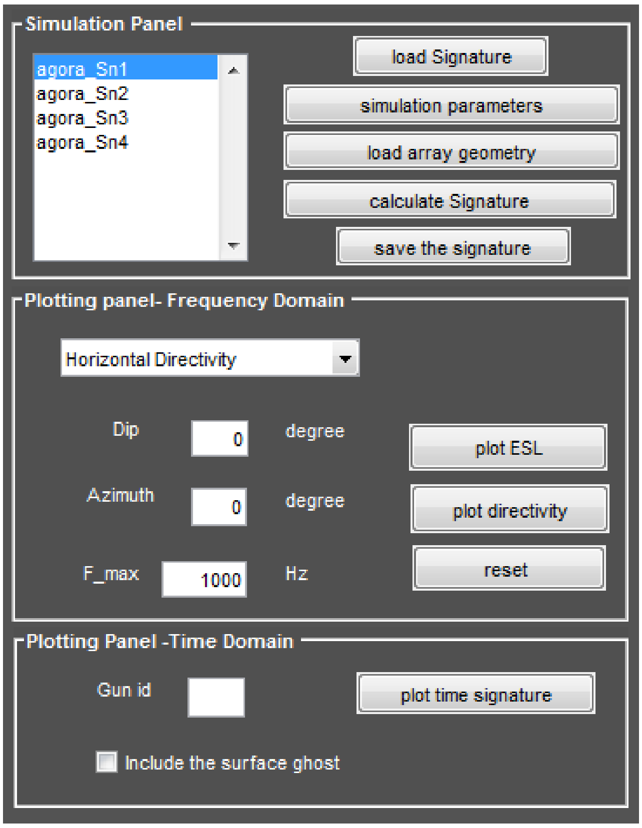

Figure 1. Three main panels of AGORA GUI

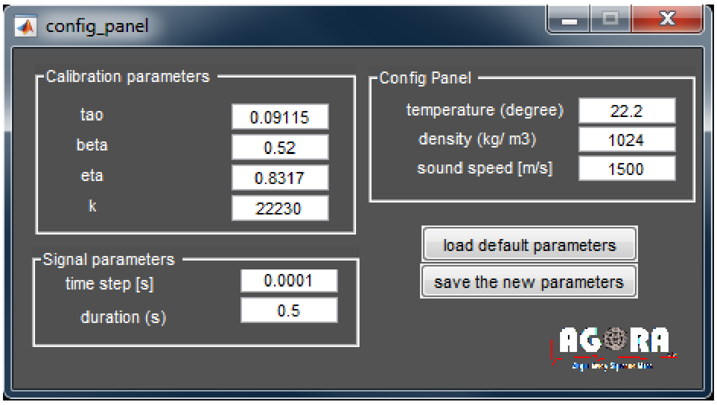

Figure 2. Configuration panel of AGORA GUI

"*Load airgun array geometry*" button opens a dialog box where you can load airgun array geometry files ".ago". These files can be created by any text editor by writing *x,y,z* coordinates, volume and firing pressure of each airguns in the array. An example *"*.ago"* file is shown below. Two examples of airgun array geometry files can be found in "…\AGORA\ago_files\" folder.

Table 2. Format of ".ago" input file*

Line 1	*2*	*% number of airguns*
Line 2	*9000*	*% receiver distance*
Line 3	*0.0 12.9 6.0 150.0 2000.0*	*% x[m] y[m] z[m] V[in3] P[firing pressure]*
Line 4	*3.0 12.9 6.0 60.0 2000.0*	*% x[m] y[m] z[m] V[in3] P[firing pressure]*

"*Calculate Array Signature*" starts the simulation after loading manually the airgun array geometry. The time domain signature of each airgun in the array is calculated. Next, you can use the plotting panels for visualizing.

"save the signature" button open a dialog box to save the calculated signatures in csv format. four files are saved as "*array_geo.aas*" (array geometry), "*freq_not.aas*" (notational frequency signatures), "*time_ghost.aas*" (time signature with surface ghost), "*time_not.aas*" (notational time signature).

3.2. Plotting panel - Frequency domain

From where you can plot the energy source level of airgun array for any direction and its directivity. You can choose vertical or horizontal directivity and the upper frequency (*Fmax*) for directivity plot. When "*plot directivity*" button is pressed, the directivity plot is generated in a new window. The plotting function is based on http://www.mathworks.com/matlabcentral/fileexchange/13200-3d-polar-plot/content/polarplot3d.m. When you click "*plot ESL*", the energy source spectral density level is plotted in the right bottom axes.

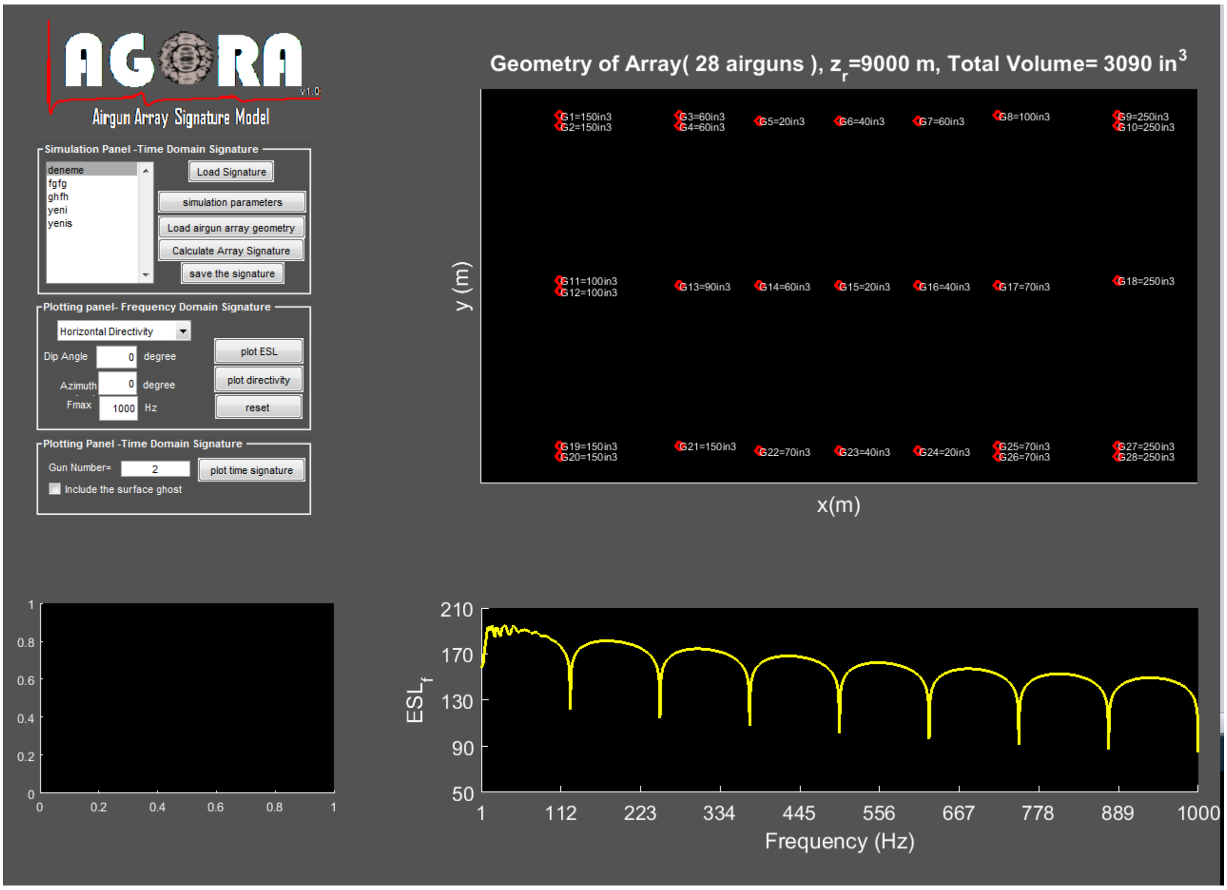

Figure 3. Plotting energy source spectral density level (in yellow color)

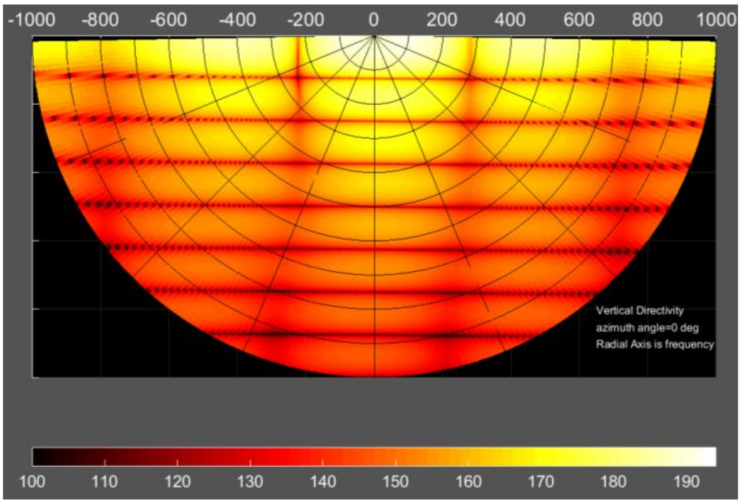

Figure 4. Plotting energy source level with vertical directivity

3.3. Plotting panel - Time domain

From this menu, you can plot the time signature of any selected airgun with or without surface ghost. Please enter the id of airgun in the box and press “plot time signature” button.

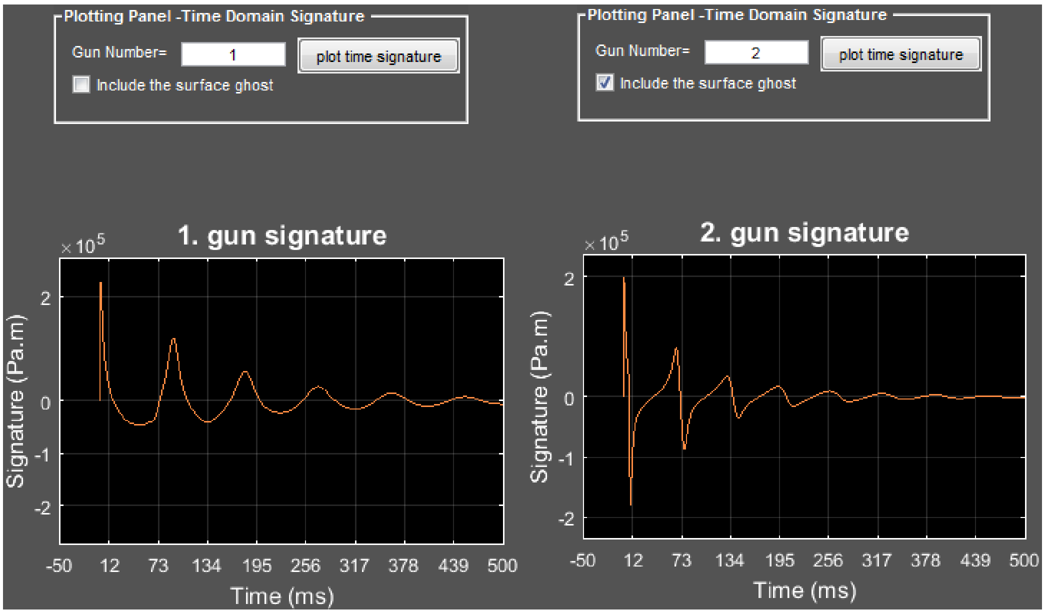

Figure 5. Plotting time domain signature without surface ghost (left) and surface ghost (right)

4. AGORA SINGLE GUN

If one desires to use the AGORA’s kernel function in order to make own implementation, “*agora_kernel*” function and “*AGORA_single_gun*” script can be used. The input format is shown by Table 3.

Table 3. The input of “agora_kernel” function.

ArrayGeo.Pg	*firing presssure [psi]*
ArrayGeo.Vg	*airgun chamber size [in3]*
ArrayGeo.zg	*firing depth [m]*
ArrayGeo.N	*Number of airguns*
ArrayGeo.x	*x location of gun [m]*
ArrayGeo.y	*y location of gun [m]*
ArrayGeo.zr	*Far field receiver depth [m]*

This scripts calls "*agora_kernel*" function which calculates the airgun signatures in different formats. Table 4 summaries the output of "*agora_kernel*" function.

Table 4. The output of "agora_kernel" function.

Signature.p_unperturbed	*Notional airgun signature without bubble interactions*
Signature.p_notional	*Notional airgun signature with bubble interactions (Bubble interactions are calculated for multiple airguns)*
Signature.p_bi_ghost	*Airgun signature with the surface ghost and bubble interactions*
Signature.F0	*Frequency vector*
Signature.P_out	*FFT of notional airgun signature with bubble interactions*

Signature.p_unperturbed and *Signature.p_bi_ghost* outputs are plotted in Figure 6.

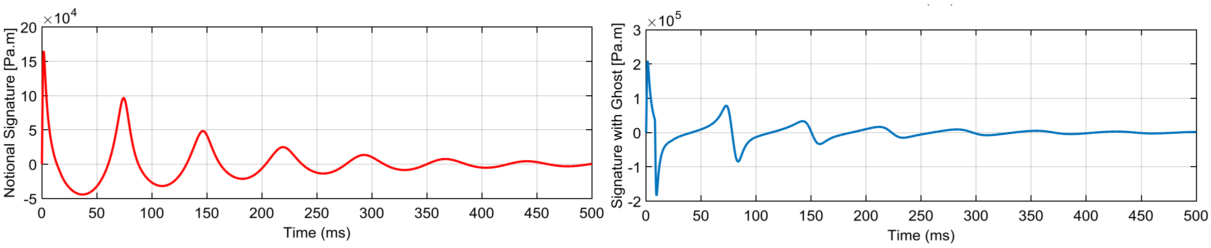

Figure 6. Plotting time domain signature without surface ghost (left) and surface ghost (right)

5. SUMMARY AND DISCUSSION

AGORA is a practical tool to calculate airgun array signatures. The calibration parameters based on multiple measurements for the various types of airgun can be replaced by user to increase the accuracy.

AGORA does not use any other model for the airgun clusters. The each airgun in the cluster is assumed as individual airgun. The bubble interactions are modelled with a perturbation approach.

The model is validated for the low frequencies up to 1 kHz. At higher frequencies, the nonlinear effects on buble motion can become more complicated and more advanced modelling techniques may require.

The developers of AGORA accepts no responsibility or liability whatsoever with regard to the results of AGORA if they are used in any commercial or scientific projects. It is a research algorithm and still needs more advanced approach to increase its accuracy for entire frequency band.

If you have any feedback or questions, please contact agora.model@gmail.com.

REFERENCES

1. H.Ö.Sertlek and M.A. Ainslie, (2015), AGORA: Airgun source signature model: its application for the Dutch seismic surveys, Conference Proceedings of UAC 2015, Crete, Greece.
2. Ziolkowski A. et al, (1982), The signature of an air-gun array: Computation from near-field measurements including interactions: Geophysics, 47(10), 1413- 1421.
3. Duren R E, 1988, A theory for marine source arrays, Geophysics 53 650–8.
4. Racca R.C. and Scrimger J.A.,(1986), Underwater acoustic source characterization of air and water guns, Tech. Rep. 06SB 97708-5-7055, DefenceResearch Establishment Pasific.
5. MacGillivray A.O., (2006), An acoustic study of seismic airgun noise in Queen Charlotte Basin, MSc Thesis.

ACKNOWLEDGEMENTS

H. Özkan Sertlek's PhD is supported by NWO-ZKO grant 'Effects of underwater noise on fish and marine mammals in the North Sea'. The measurements of single airgun data are supplied by the Joint Industry Programme on Sound and Marine Life. The author thanks Michael A. Ainslie for many discussions about the source&propagation modelling and their metrics; Alex MacGillivray for his willingness to answer our questions about his MSc thesis during the development of this model; Robert Laws for many stimulating discussions about airgun source modelling; Mark Prior and Stefania Giodini for their comments and suggestions on during testing AGORA; Mustafa Ünal and James Campbell for their feedback on MATLAB GUI and Michael B. Porter for his help for distributing AGORA from the Ocean Acoustic Library webpage.