NAVAL
POSTGRADUATE
SCHOOL

MONTEREY, CALIFORNIA

THESIS

DEVELOPMENT OF FOUR-ELEMENT END-FIRE ARRAY AS SEISMO-ACOUSTIC SONAR SOURCE

by

Steven E. Rumph

September 2003

Thesis Advisor:	Steven R. Baker
Co-Advisor:	Thomas G. Muir

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2003	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE: Development of Four-Element End-fire Array as Seismo-Acoustic Sonar Source	**5. FUNDING NUMBERS**
6. AUTHOR(S) Steven E. Rumph	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

Because the inherent dangers of buried mines still possess the ability to deny or severely limit the projection of naval forces ashore, continued research and development of landmine detection capabilities is paramount. One promising method for buried mine detection, still in the early development stages, is the use of Seismo-Acoustic waves for target echo location.

The goal of the research described in this thesis was to design and develop a four-element end-fire array to be used as a Seismo-Acoustic SONAR source to preferentially excite Rayleigh waves. A four-element end-fire array, consisting of four vertical-motion inertial mass shakers spaced 25 cm apart (approximately one-quarter wavelength), was deployed on the sand in the near-surf zone of Del Monte Beach. The beam of the array was directed to end-fire by using a delay/pulse generator to individually trigger four function generators. Testing at the beach was conducted and results have shown nice end-fire beam patterns at a radius of 3.5 meters with approximately 15 dB suppression to the rear of the array relative to the forward direction.

14. SUBJECT TERMS Seismic SONAR, Seismo-Acoustic, Landmine Detection, Seismic Waves, End-Fire Array	**15. NUMBER OF PAGES** 107
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

THIS PAGE INTENTIONALLY LEFT BLANK

DEVELOPMENT OF FOUR-ELEMENT END-FIRE ARRAY AS SEISMO-ACOUSTIC SONAR SOURCE

Steven E. Rumph
Lieutenant, United States Navy
B.E.E., Georgia Institute of Technology, 1994

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING ACOUSTICS

from the

NAVAL POSTGRADUATE SCHOOL
September 2003

Author: Steven E. Rumph

Approved by: Steven R. Baker
Thesis Advisor

Thomas G. Muir
Co-Advisor

Kevin B. Smith, Chairman
Engineering Acoustics Academic Committee

ABSTRACT

Because the inherent dangers of buried mines still possess the ability to deny or severely limit the projection of naval forces ashore, continued research and development of landmine detection capabilities is paramount. One promising method for buried mine detection, still in the early development stages, is the use of seismo-acoustic waves for target echo location.

The goal of the research described in this thesis was to design and develop a four-element end-fire array to be used as a seismo-acoustic SONAR source to preferentially excite Rayleigh waves. A four-element end-fire array, consisting of four vertical-motion inertial mass shakers spaced 25 cm apart (approximately one-quarter wavelength), was deployed on the sand in the near-surf zone of Del Monte Beach. The beam of the array was directed to end-fire by using a delay/pulse generator to individually trigger four function generators. Testing at the beach was conducted and results have shown nice end-fire beam patterns at a radius of 3.5 meters with approximately 15 dB suppression to the rear of the array relative to the forward direction.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I would like to thank Professor Steve Baker for his excitement and motivation, which kept the fire lit throughout the entire process. Next, I would like to thank Sam Barone and George Jaksha for all of their electrical and mechanical experience in addition to their knack for being genuinely friendly and helpful. I would also like to thank ENS Douglas MacLean for his help at the beach. Lastly, I would like to thank Ron Cowen and Keith Wyckoff for their help in keeping the Gator ready for the beach.

I. INTRODUCTION

A. RESEARCH MOTIVATION

Landmines deny access to territory and can cause huge changes in battle plans. They are cheap to own, but very expensive to counter. From a naval perspective, they have the overall effect of severely limiting the projection of naval forces ashore. Continued research and development of landmine detection capabilities is paramount. Current methods for detection of buried mines are very inefficient. The use of metal detectors, which require the operator to be within the kill radius of the mine, and mine hunting dogs, which are very limited in their usefulness, just aren't sufficient to effectively reduce the risk to troops whose mission is to take or occupy territory that may contain buried landmines.

The search for a more efficient method has been ongoing. Dr. Thomas Muir initiated the concept of using seismo-acoustic SONAR to generate waves which travel in the surface region of the ground to be used for buried landmine detection. Figure 1 depicts the seismo-acoustic SONAR concept. Seismic waves are artificially generated, travel through the surface region, get reflected off buried objects, and then are received by vibration sensors to determine range and bearing to the object.

Guided seismo-acoustic interface (e.g. Rayleigh) waves have especially desirable characteristics for this application. They are localized near the interface, which, for the wave speed and the frequency used in this research is approximately one meter of depth. Also, their propagation from a compact source is cylindrical. As a result, the wave intensity on targets of interest is enhanced and the reverberation suppressed, compared to a bulk wave.

The use of seismo-acoustic SONAR for target location and identification is still in the early development stages. Preferential excitement of guided surface seismo-acoustic waves and the use of a method of signal processing called vector polarization have shown very positive results. Exploration of new innovative source and receiver designs is the key which will unlock many doors as this technology continues to develop.

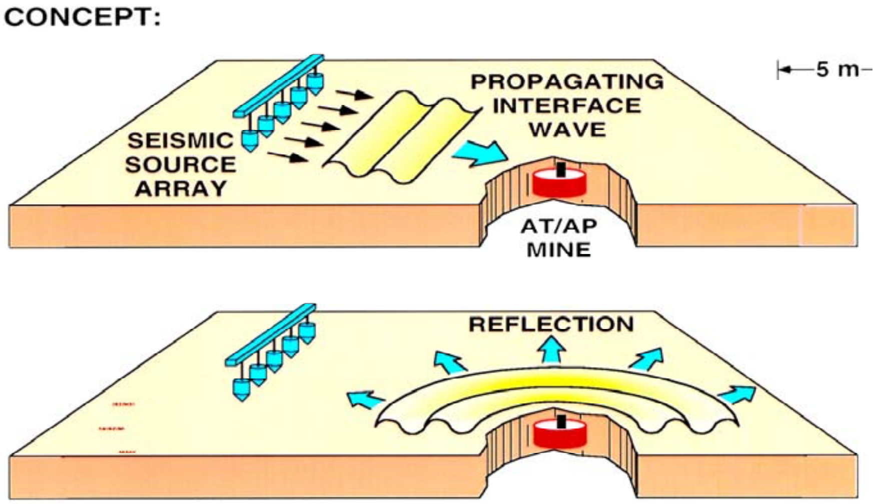


Figure 1. Seismic SONAR Excited Waves

B. RESEARCH OBJECTIVE

The goal of the research described in this thesis was to design and develop a four-element end-fire array to preferentially excite Rayleigh waves in the sand area of the beach, to be used as a seismo-acoustic SONAR source. An end-fire source array, consisting of four vertical-motion inertial mass shakers, was deployed on the sand above the surf zone at Del Monte Beach. Measurements were made of the radiated vertical and radial ground motion, at various ranges and angular positions, for various transient signals and element-to-element time delays. These measurements were used to determine Rayleigh wave propagation speed, signal attenuation versus range, and array beam patterns.

II. PRIOR RESEARCH

Seismo-acoustic SONAR research was initiated by Dr. Thomas Muir while he was at the Applied Research Laboratories of the University of Texas at Austin in the early 1990's. Since then, Dr. Muir has worked at the Naval Postgraduate School and is now working at the University of Mississippi.

Several students at the Naval Postgraduate School have contributed to this effort. LT Fred Gaghan developed a two degree-of-freedom source designed to preferentially excite Rayleigh waves. [Ref. 6] LT Sean Fitzpatrick developed stronger, vertical-motion sources. Using two of these sources side-by-side, he was able to locate targets at ranges up to five meters. [Ref. 5] MAJ Patrick Hall worked together with Fitzpatrick, but his research concentrated on target strength. His thesis showed that adding mass to the target caused an increase in the target strength. [Ref. 7] In 2000, CPT Kraig Sheetz continued the seismic SONAR work by developing a seven-element linear array, attaining a SNR of 9dB for the target reflections. [Ref. 11] In 2002, LT Scott McClelland developed a seismic SONAR source capable of rolling. His source consisted of shakers mounted inside a cylinder. [Ref. 9] The most recent work was done by ENS Douglas MacLean in 2003. He used a small, tracked robot with vertical-motion shakers mounted on top. [Ref. 8]

THIS PAGE INTENTIONALLY LEFT BLANK

III. GENERAL OVERVIEW OF SEISMIC WAVES

The foundations for the study of seismic waves are based primarily on a working knowledge of elasticity, stress and strain, and various solutions of the wave equation. The mathematics and derivations required to fully cover these topics are the subject of many textbooks. This thesis does not attempt to cover those topics in depth, but merely to present the reader with a general overview of some of the applicable results without getting lost in the complex mathematics.

For the purposes of this thesis, it will be assumed that the sandy area of the beach used for testing is a medium which has the characteristics of being isotropic, homogeneous, and elastic. That is to say, the properties of the beach do not depend on direction, the beach is uniform throughout and the properties can be considered to be the same at different points, and when it is displaced and deformed by applied forces, every point in the beach will return exactly to its original position when the forces are removed. In truth, the mechanical properties of the beach vary with depth due to, for example, the overburden of sand above that at a certain depth, or the amount of water saturation, so it is not a homogeneous, isotropic medium. This will have to be addressed at some point in the future.

The beach also has another characteristic of interest. It is considered to be a half-space since there is an interface between the earth and the air, or the earth and a free surface. If the beach were simply an infinite medium with no interface, then the only waves that would travel would be **body waves**, consisting of compressional waves and shear waves. But since there is an interface with the air, another variety of waves, called **surface waves**, propagate in the vicinity of the surface-air interface, but decay rapidly with depth.

A. BODY WAVES

As mentioned above, there are two types of body waves: compressional waves and shear waves. Compressional waves travel the fastest and are therefore also known as P-waves (P for primary). Since the particle motion for a compressional wave is in the direction of travel, they are also sometimes referred to as longitudinal waves. The P-wave travels with velocity given by

$$V_p = \sqrt{\frac{k + 4\mu/3}{\rho}}$$

where k is the bulk modulus, μ is the shear modulus, and ρ is the density of the medium. [Ref. 13] Figure 2 below shows the longitudinal propagation of a P-wave.

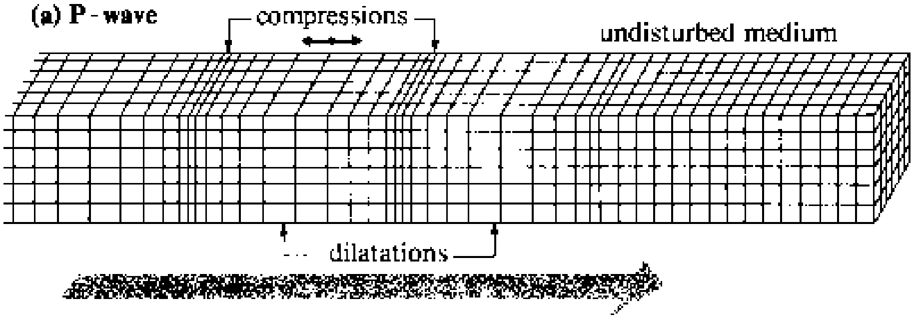


Figure 2. Propagation of a Compressional Wave [from Ref. 2]

The second type of body wave is the shear wave. They propagate more slowly than the P-wave and are referred to as S-waves (S for secondary or shear). The particle motion of a shear wave is perpendicular to the direction of travel. Because of the perpendicular particle motion, shear waves are also called transverse waves. They may be polarized to either vertical or horizontal orientations. When this is the case, they are called “SV” or “SH” waves, respectively. The S-wave travels with velocity given by

$$V_s = \sqrt{\frac{\mu}{\rho}}$$

where V_s is the velocity, μ is the shear modulus, and ρ is the density. [Ref. 13] Figure 3 below shows the transverse propagation of an S-wave.

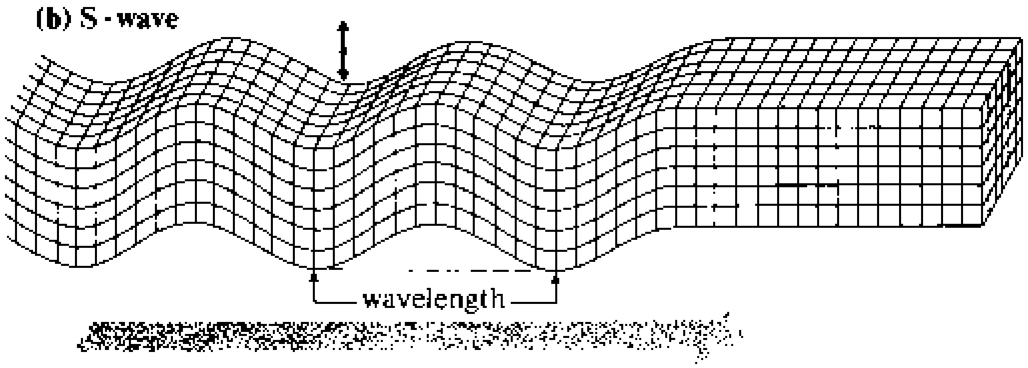
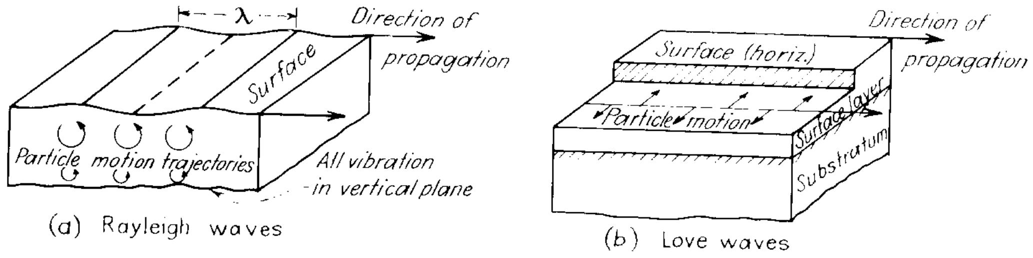


Figure 3. Propagation of a Shear Wave [from Ref. 2]

In general, body waves are not the desired type of waves to be used in seismic mine detection because they decay more quickly with range than do surface waves and because they will reflect from interfaces between subsurface layers and inhomogeneities. Consequently, there will be numerous reflected rays arriving at sensors, making it very difficult, if not impossible, to differentiate between the initial drive signal, the bottom bounce reflection signals, and the return signal from the target. Both types of body waves attenuate proportional to 1/r within the ground. Along the surface, body waves attenuate proportional to $1/r^2$. [Ref. 6]

B. SURFACE WAVES

There are two types of surface waves which can propagate: Love waves and Rayleigh waves. Figure 4 shows the two types for a side-by-side comparison.



(a) Rayleigh waves

(b) Love waves

Figure 4. Characteristics of (a) Rayleigh Waves, and (b) Love Waves Traveling along the Surface of a Solid [from Ref. 3]

Love waves are essentially ducted SH-waves which attenuate with depth. The particle motion of Love waves is horizontal and perpendicular to the direction of propagation. This type of wave does not exist in a homogeneous, isotropic half-space since they require layers.

Rayleigh Waves are surface waves of the P-SV type and were first described by Lord Rayleigh in 1885. Figure 5 below shows propagation of a Rayleigh wave.

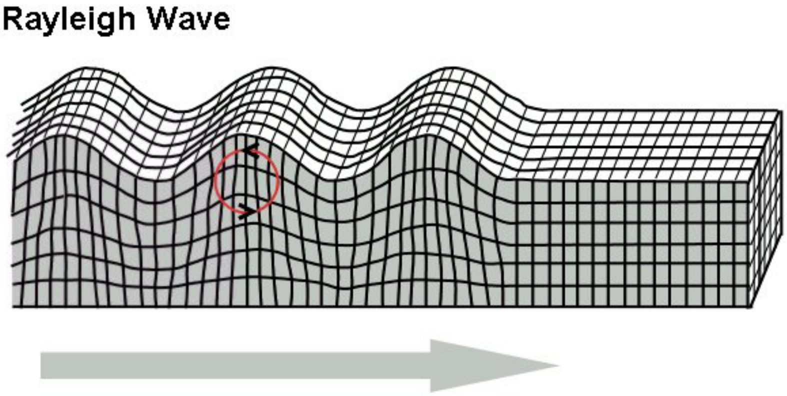


Figure 5. Propagation of a Rayleigh Wave [from Ref. 2]

They are sometimes referred to as ground roll and have some unique characteristics which make them optimal for seismic mine detection.

- They propagate at a speed slightly less than shear speed.
- The particle motion is an elliptical trajectory which is retrograde at the surface.
- They are non-dispersive in a homogeneous half-space.
- Particle motion is localized to a layer approximately two wavelengths deep.

Rayleigh waves propagate with velocity: $c_r = c_s \kappa$, where c_r is the Rayleigh wave speed, c_s is the shear wave speed, and κ is a dimensionless parameter, given by:

$$\kappa^6 - 8\kappa^4 + (24 - 16\gamma^2)\kappa^2 + (16\gamma^2 - 16) = 0,$$

$$\text{with} \quad \gamma^2 = \frac{c_s}{c_l} = \frac{\mu}{\lambda + 2\mu} = \frac{1 - 2\nu}{2 - 2\nu},$$

where γ is a dimensionless parameter, c_s is the shear wave speed, c_l is the longitudinal wave speed, λ and μ are the Lame constants, and ν is Poisson's Ratio. [Ref. 5] For most solid/fluid boundaries (including sand/air), κ is approximately 0.9. [Ref. 5]

The elliptical trajectory is very significant in that it allows easy identification, a big advantage of the Rayleigh wave. Shown below are the two modes of elliptical propagation.

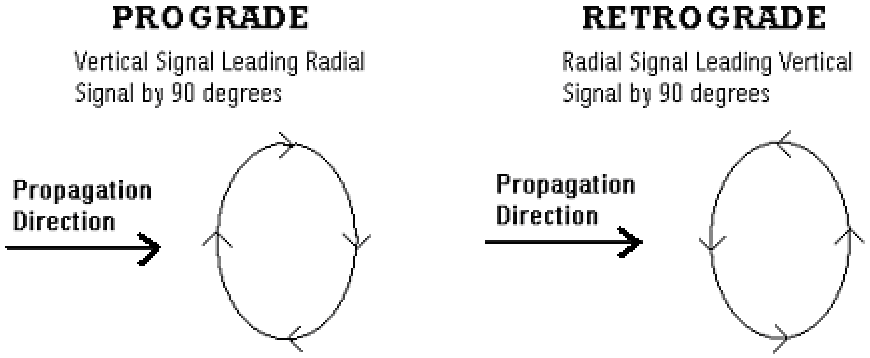


Figure 6. Prograde and Retrograde Elliptical Propagation.

Figure 7 shows the horizontal (U) and vertical (W) displacements for Rayleigh waves in a homogeneous half-space. The elliptical trajectory is retrograde for depths less than h, and then switches to prograde for depths greater than h. Being localized to a layer of approximately two wavelengths deep is a key advantage of using Rayleigh waves to detect buried objects. Since they decay exponentially with depth, there is little interference due to complex substructure or bottom reflections.

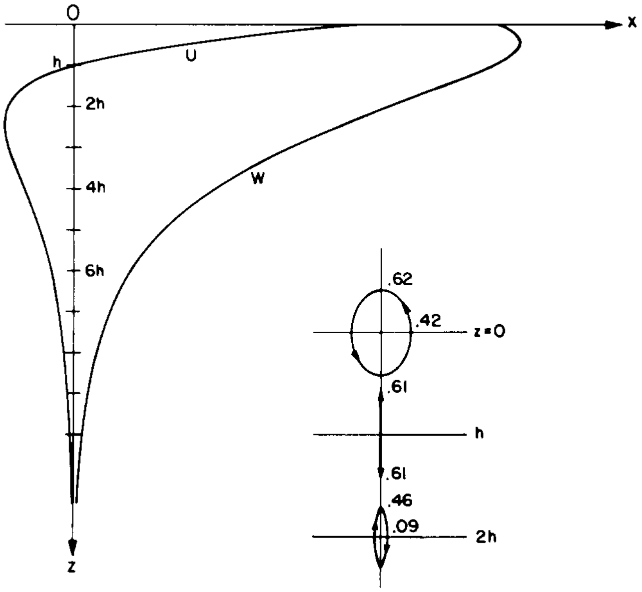


Figure 7. The horizontal (U) and vertical (W, down) displacements for Rayleigh waves in a homogeneous half-space. U vanishes at depth h. The path of the particles is elliptic retrograde for $z < h$ and elliptic direct (prograde) for $z>h$. [from Ref. 1]

As mentioned above, the amplitude of the body waves decreases in proportion to $1/r$ (and even $1/r^2$ along the surface), whereas the amplitude of the Rayleigh wave decreases in proportion to $1/\sqrt{r}$ so that beyond about 2.5 wavelengths from the source, Rayleigh waves carry about 67% of the seismic energy at the surface. [Ref. 1] It is quite obvious that Rayleigh waves are the waves of choice for seismo-acoustic SONAR mine detection.

IV. EQUIPMENT

A. LISTING

Power Generator: Honda EU Inverter 2000i, Gasoline Powered, 60 Hz, 120 V, 13.3 A, 1.6 kVA rated output

Delay/Pulse Generator: Stanford Research Model DG535 Four Channel Digital

Function Generator: Four Hewlett Packard 3314A's

Driver: Two Carvin 1000 Watt Power Amps, Model DMC 1000

Source: Four AURA Bass Shakers, Model AST-1B-4

Receiver: Seismometers composed of SM-11 Geophones (SENSOR Nederland products) connected to Seismometer Receiver Panel

Oscilloscopes: Tektronix TDS 3014 and TDS 3034, Four Channel Color Digital Phosphor

Accelerometer: Model J353B03, S/N 9319, Voltage sensitivity 10.24 mV/g

Accelerometer Amplifier: Model PCB 482A17

Transportation to Beach: John Deer Trail Gator and Physics Trailer

B. DESCRIPTION

For use at the beach testing site, the equipment was mounted in the physics trailer, and then towed to the beach using the John Deere Trail Gator. The equipment, as mounted in the physics trailer, is shown in Figure 8 below.

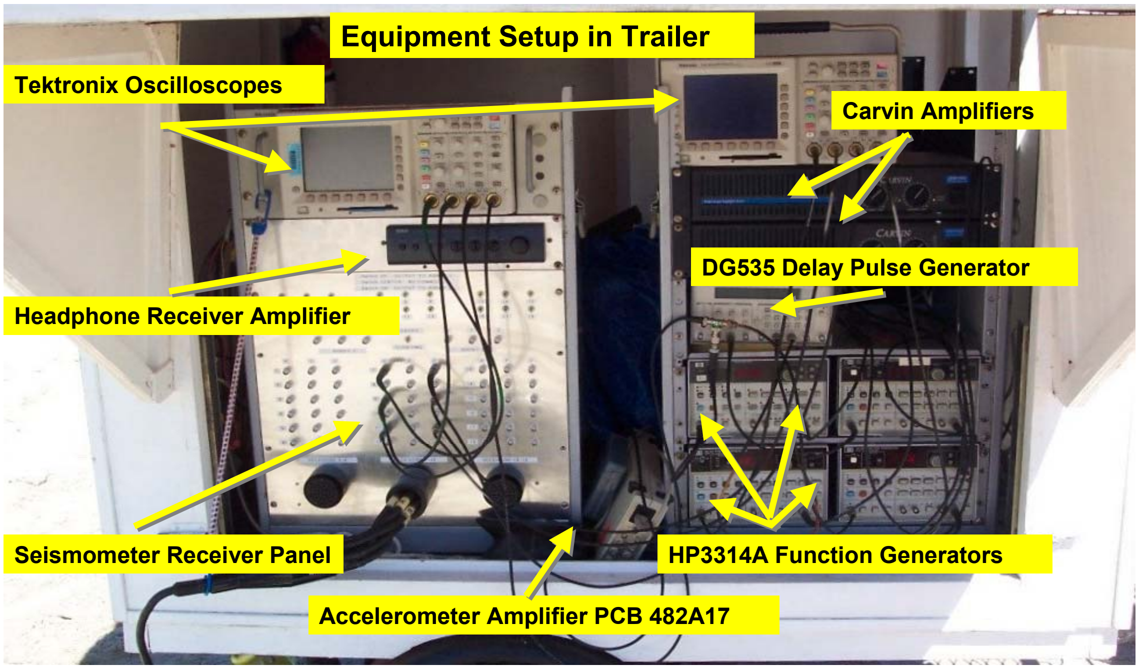


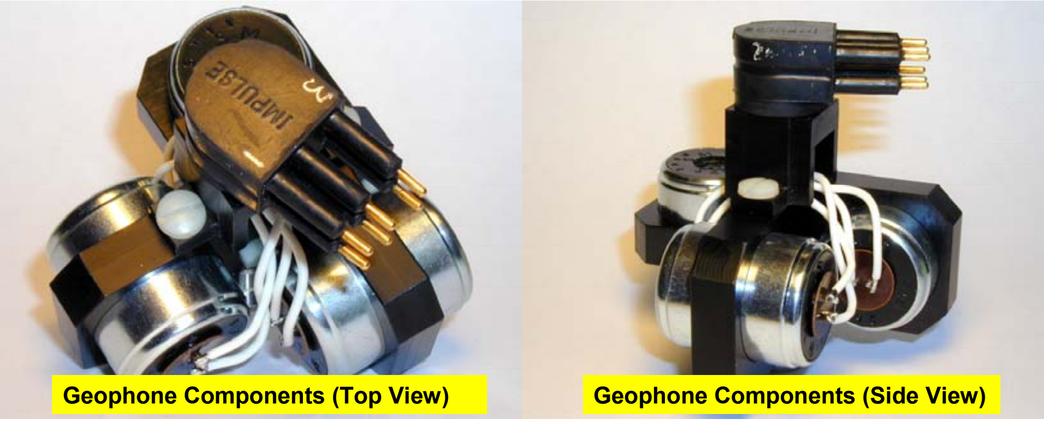
Figure 8. Beach Testing Equipment (Mounted in Trailer)

The power supply used at the beach was the Honda EU 2000i generator. When set on the ground, the generator produced enough vibrations to cause noise in the seismometer output signals. To prevent interference from these vibrations, the generator was set on a plywood mount on top of the front of the John Deere Trail Gator. The Stanford Research DG535 Delay/Pulse Generator was used to externally trigger the function generators to provide a time delay between each element of the array. It was also used to trigger the oscilloscopes to provide more accurate measurements. The HP 3314A Function Generators were used to generate various transient wave forms to be used to drive each of the four source elements. The Carvin Power Amps were used to amplify the signal from the function generators and output the signals to drive the source elements. Four AURA Bass Shakers were used to implement a four-element end-fire array. SM-11 Geophones were potted in watertight cylinders and used to measure the output of the source elements. Once potted, the geophones were then referred to as seismometers. The outputs of the seismometers were connected via the seismometer receiver panel to Tektronix oscilloscopes which were used to view and store output from the seismometers and accelerometers.

V. RECEIVER DEVELOPMENT

A. SEISMOMETERS

The seismometers are each composed of three SM-11 Geophones (SENSOR Nederland products) mounted in X-Y-Z coordinates and potted inside a watertight cylinder of height 6 cm and diameter 9.5 cm. The internal components and the final potted version of the seismometer receiver are shown in Figures 9 and 10. Specifications for SM-11 geophones are given in Appendix A.


Geophone Components (Top View)

Geophone Components (Side View)

Figure 9. SM-11 Geophones mounted in X-Y-Z coordinates

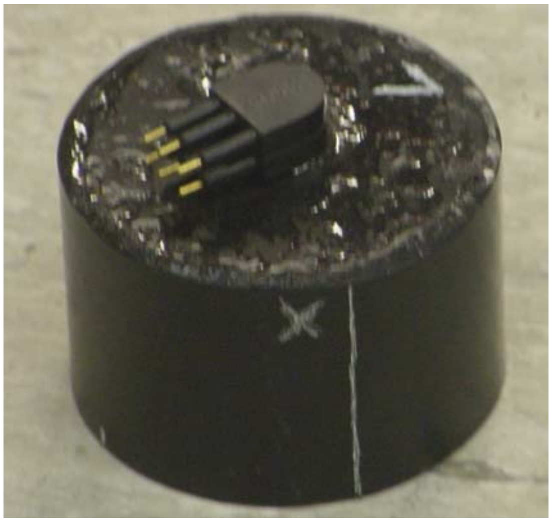


Figure 10. Final potted version of Seismometer

Initial testing of the SM-11 geophones for use in Seismic SONAR applications was conducted by Cpt. Kraig Sheetz, and it was concluded that the vertical and horizontal models of the SM-11 could be oriented at any angle and still adhere to the advertised specifications. [Ref. 11]

Once the geophones were potted, testing was conducted to conclusively determine the X-Y-Z coordinates of each seismometer. The author constructed a pendulum calibration device consisting of a threaded rod with a weight attached to the bottom. The side and top views of the pendulum device are shown in Figure 11. The rod was pulled to the back-stop against a guide to ensure the same starting position prior to release. With sensor outputs ranging from 0 mV to approximately 400 mV, the pendulum device produced a repetitive, consistent output within 3 mV. Seismometer output pins 1 & 2, and 3 & 4 were used to monitor the outputs for the X and Y coordinates, respectively. Each seismometer was tested by striking the seismometer and observing the output several times to verify consistency, and then repeating in 10° increments. The maximum X and Y outputs and their respective coordinates were recorded. The resulting X and Y coordinates were then etched onto the seismometers with a soldering iron and then painted white so they could be easily seen during field testing. This etching can also be seen in Figure 10 showing the potted version of a seismometer.

Figure 11. Seismometer Pendulum Calibration Device (side and top views)

After determining the proper X-Y-Z coordinates of the seismometers, testing at the beach was conducted to determine the sensitivity difference among the individual seismometers. Each seismometer was placed at the same location, two meters from the center of the end-fire array, one at a time, and the radial and vertical signals were recorded for a constant source input value. Using the cross-correlation function in MATLAB, calibration values were calculated. To give each seismometer a constant amplitude with reference to seismometer number one, each seismometer output signal is divided by its individual calibration value. This allowed the beam pattern data, taken using six seismometers, to be consistently constructed relative to seismometer number one. The setup for this is depicted in Figure 12 below. The calculated calibration values for the radial and vertical outputs for each of the six seismometers are shown below in Table 1.

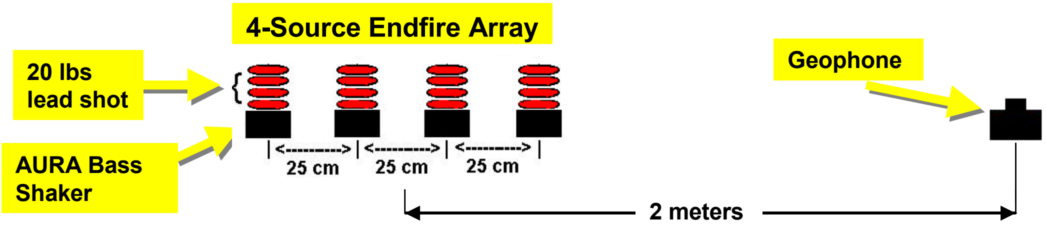


Figure 12. Seismometer Calibration Setup

Seismometer	Radial Calibration Value	Vertical Calibration Value
1	1.0000	1.0000
2	1.0097	1.0685
3	0.8666	1.0603
4	0.9159	1.0670
5	1.0165	1.0669
6	0.8522	1.0469

Table 1. Calibration Values for Seismometers 1-6.

B. SEISMOMETER RECEIVER PANEL

The seismometer receiver panel was designed to allow for eighteen seismometers to be used at once. There are three large 36-pin connectors at the bottom of the panel. Each connector accommodates six seismometers. The panel gives the operator the option (via a three pole switch) to place each seismometer onto one of two arrays, or to be used singularly. Each array is composed of the selected seismometers connected in parallel. The array outputs are intended to be input to a current amplifier. The outputs of each array (X, Y, Z) are accessed via BNC connectors. The outputs of the individual seismometers also have BNC connectors so that both the arrays and individual seismometers may be simultaneously monitored. Each seismometer has six connector pins. Pins one and two are for the X coordinate (positive and negative respectively); pins three and four are for the Y coordinate (positive and negative respectively); pins five and six are for the Z coordinate (positive and negative respectively). Once plugged into the receiver panel, the negative pins are all connected in parallel via a bus bar (ground). The operator has the option to select either a floating or chassis ground. The design and testing were performed by the author, and the actual construction was the handiwork of electronics technician, Sam Barone. Shown below as Figure 13 is a simplified schematic depicting the connections for a single six-pin seismometer. Figure 14 shows the front and rear views of the completed receiver panel.

The receiver panel worked well and facilitated the setup and data recording at the beach. The only problem noted was that some of the 36-pin connectors made poor (presumably high resistance) contact, causing capacitive-coupled electrical crosstalk between the source output and the seismometer output. Care was taken to ensure that the connections were clean and fitted snugly, and the electrical crosstalk problem fixed itself.

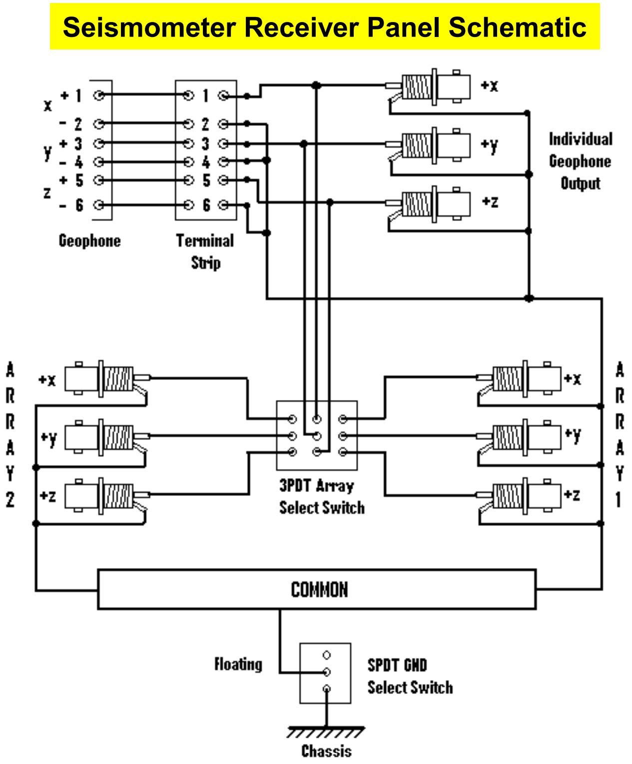


Figure 13. Schematic for Single Seismometer.

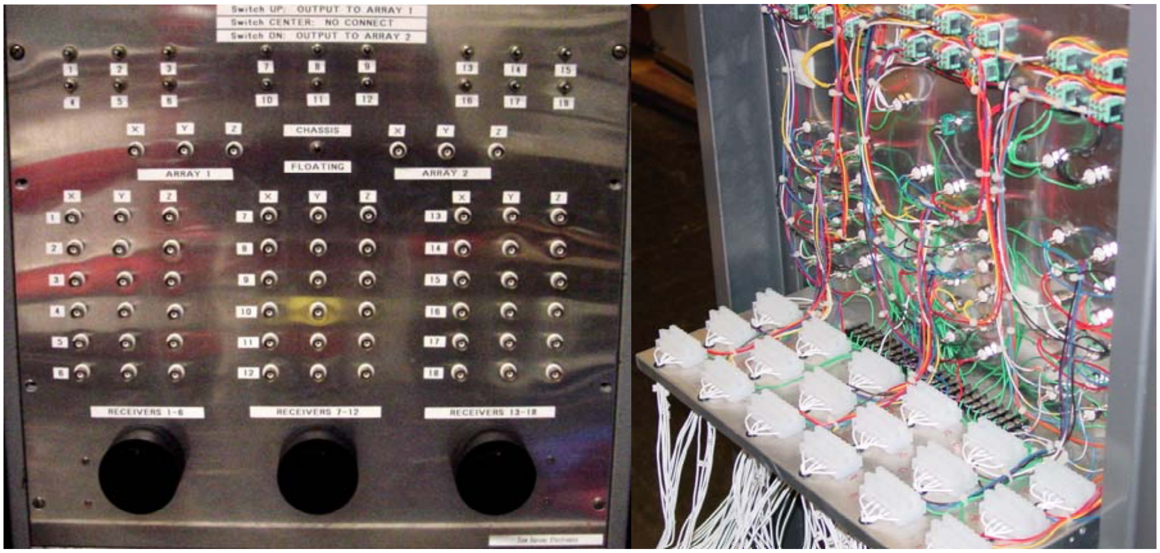


Figure 14. Seismometer Receiver Panel Front and Rear Views

THIS PAGE INTENTIONALLY LEFT BLANK

VI. SEISMO-ACOUSTIC SOURCE DEVELOPMENT

A. END-FIRE ARRAY THEORY

The four-element end-fire seismo-acoustic source array is a linear array of identical, equally spaced elements whose collective beam is steered to end-fire. Steering of the array is accomplished by applying the appropriate time delay between the drive signals of individual elements. The time delay, $\mathbf{t_d}$, is dependent upon the wave speed, **c**, and the spacing between the elements, **d**, according to $\mathbf{t_d} = \mathbf{d}/\mathbf{c}$.

From previous research, as well as the present investigation, the Rayleigh wave speed was known to be approximately 100 m/s at the beach test site. Also, based on previous research, the nominal operating frequency for the transient signals employed was chosen to be 100 Hz. To best suppress P-waves in particular, and to selectively excite Rayleigh waves using vertical-motion sources, four sources were employed, spaced one-quarter wavelength apart, with their outputs delayed by one-quarter period between elements. For a nominal frequency of 100 Hz and a wave speed of 100 m/s, **d** = 0.25 meters and $\mathbf{t_d}$ = .0025 sec.

To determine the theoretical beam pattern of the array, first it must be established where are the Fresnel (near field) and Fraunhofer (far field) region boundaries. The following equation relates these distances:

$1.356R <$ Near Field $< \pi R^2/\lambda <$ Far Field. [Ref. 15]

For a linear array, the value of R is one half the length of the array. For an array of four Aura Bass Shakers spaced 25 cm apart, L = 0.88m and R = 0.44m, so that

0.5966m < Near Field < 0.6082m < Far Field.

The distance is measured from the center of the array; the distance to the outside edge of the array is 0.44 meters. With the far-field region beginning at a distance of 0.61 meters, for all practical purposes of this array, we will be operating in the far field.

For comparison to theory, the array was assumed to be composed of identical point sources. For a four-element, linear array of identical, equally spaced elements, the normalized directional factor was computed as

$$S_4 = (1/4)\left[e^{j(ux1 + \phi 1)} + e^{j(ux2 + \phi 2)} + e^{j(ux3 + \phi 3)} + e^{j(ux4 + \phi 4)}\right].$$

For **d** = **0.25** meters, x1 = -0.375, x2 = -0.125, x3 = 0.125, x4 = 0.375.

For end-fire steering, $\phi 1 = 0$, $\phi 2 = \pi/2$, $\phi 3 = \pi$, $\phi 4 = 3\pi/2$.

B. SOURCE INPUTS

Three types of input signals were employed to drive the sources. The first was one cycle of a 100 Hz sine wave (Figure 15). This signal was used during original ground coupling testing in the lab. Function generator settings were as follows: N cycle, Freq = 100 Hz, Amptd = 1.41V (peak; corresponding to 1 V rms), Offset = 0, Sym = 50%, Phase = 0, N = 1, Trig = Ext leading edge. The output of the power amplifier was adjusted to produce 80 Vpk-pk and can be seen in Figure 16. The power amplifier is ac-coupled, and does not pass dc. The consequence of this can be seen in all of the figures which follow showing the amplifier output (Figures 16, 18, and 20).

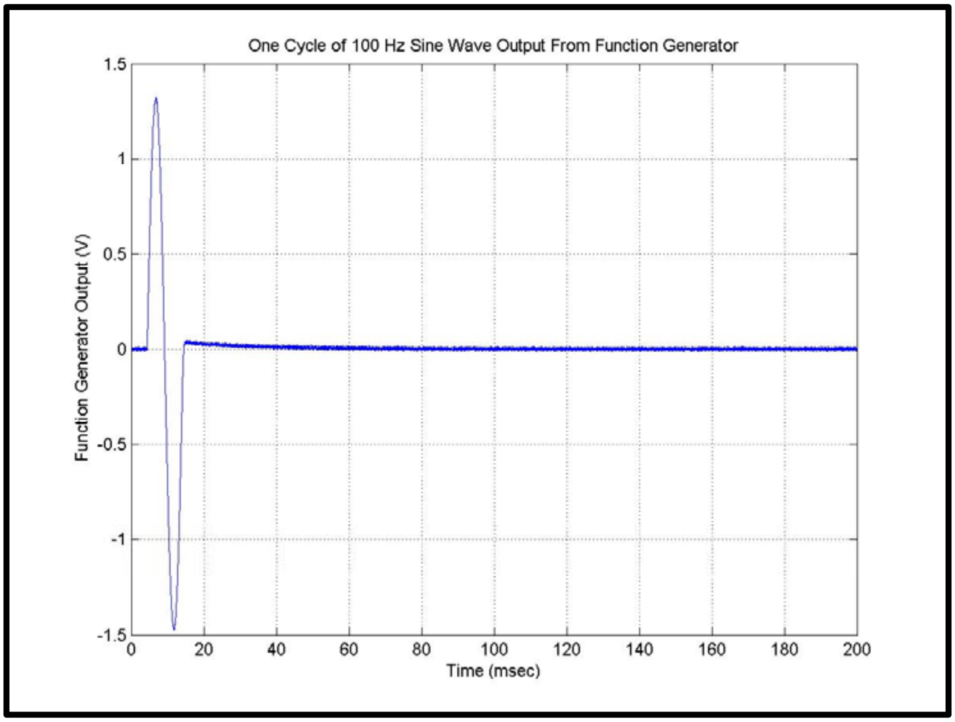


Figure 15. Function Generator Output for 100 Hz Sine Wave

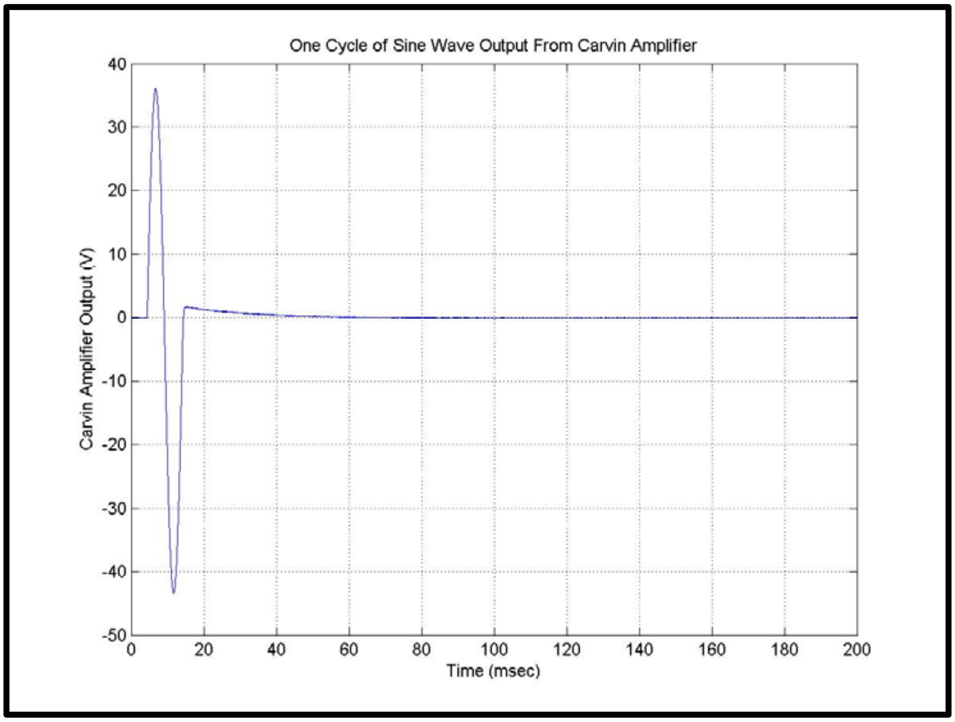


Figure 16. Carvin Amplifier Output for 100 Hz Sine Wave

The second type of signal used was a five-cycle burst of a 100 Hz sine wave (Figures 17 and 18). This signal was used a great deal in determining both the wave speed and the beam pattern of the array. Having five cycles made the cross correlation and the wave speed calculations more accurate.

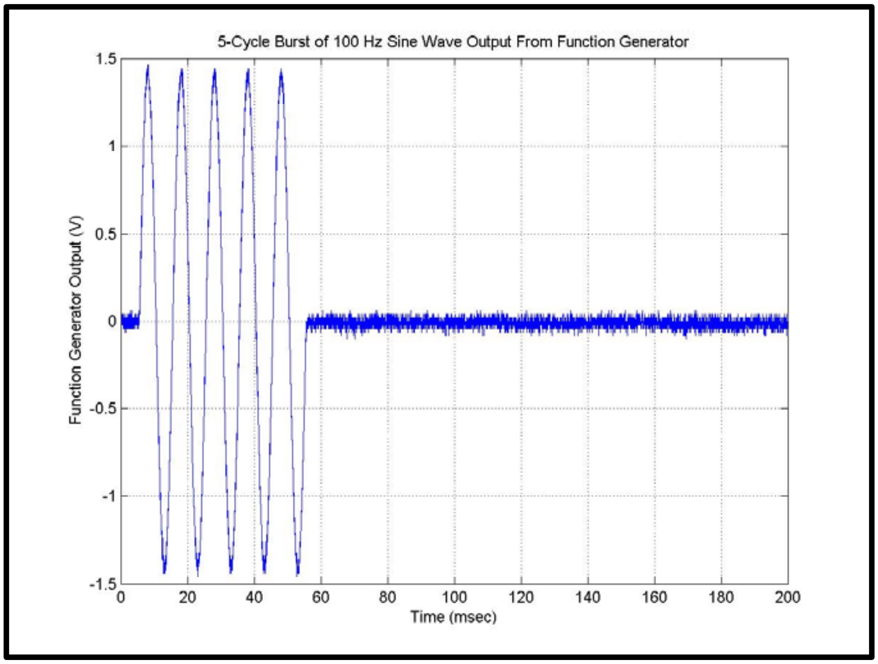


Figure 17. Function Generator Output for 5-Cycle Burst of 100 Hz Sine Wave

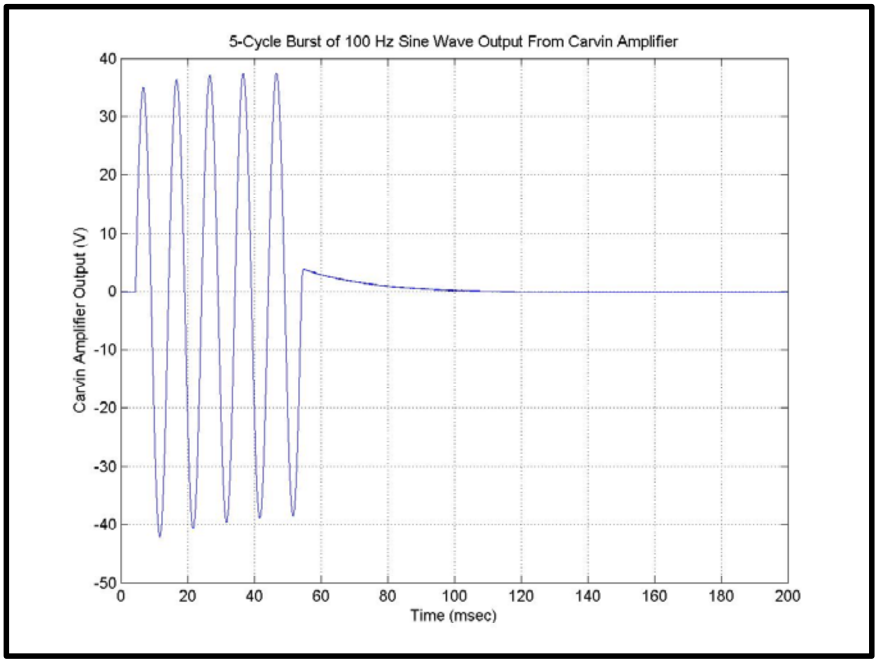


Figure 18. Carvin Amplifier Output for 5-Cycle Burst of 100 Hz Sine Wave

The third, and last, type of signal used was one cycle of a 100 Hz haversine wave (Figures 19 and 20). The haversine wave was produced by shifting the phase of the sine wave by 90 degrees and then adjusting the offset to set the baseline to zero. The function generator settings were as follows: N cycle, Freq = 100 Hz, Amptd = 1.41V, Offset = 0.83V, Sym = 50%, Phase = -90, N = 1, Trig = Ext leading edge. The output of the Carvin Amp was adjusted to produce an output of approximately 70 Vpk-pk, a setting just below the point at which the output signal began clipping. The Carvin Amp was ac coupled so that it did not pass dc, and thus the signal produced was not a perfect haversine wave as can be seen in Figure 20. This unipolar signal was designed to produce positive vertical acceleration of the shaker sources, and thus to produce a predominantly downward-directed force applied to the ground. The haversine wave also provides a smooth transition as the signal excites the shaker.

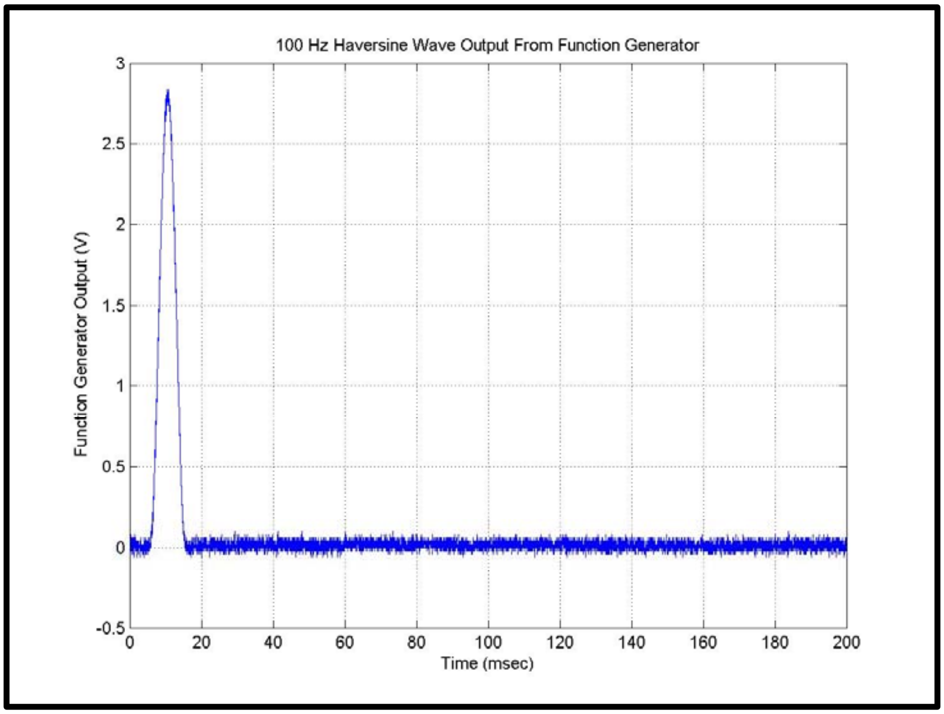


Figure 19. Function Generator Output for 100 Hz Haversine Wave

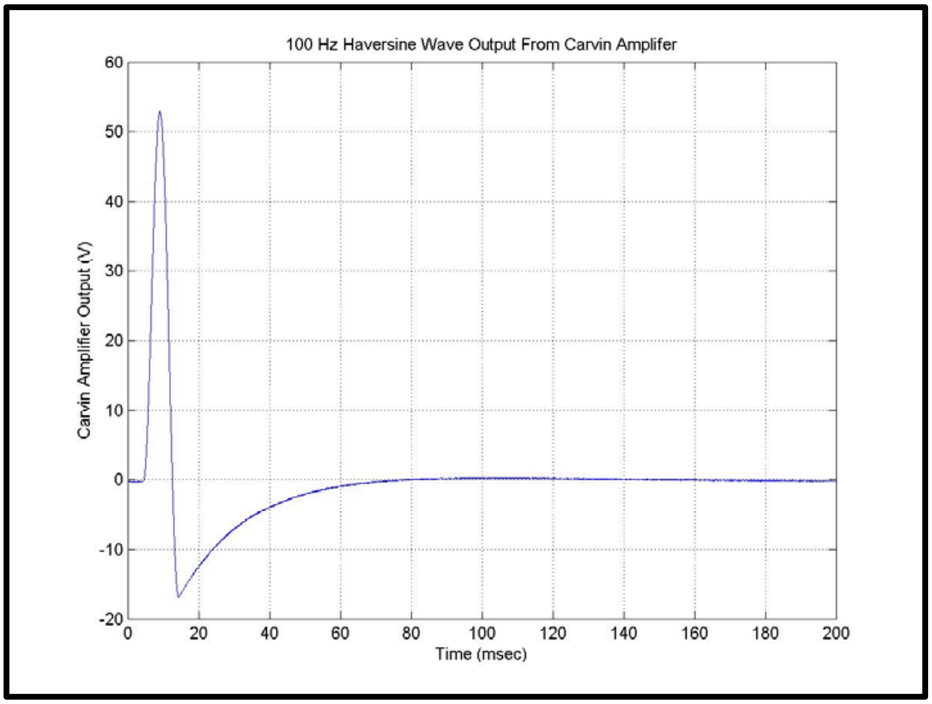


Figure 20. Carvin Amplifier Output for 100 Hz Haversine Wave

C. GROUND COUPLING

The inertial shaker sources rest on the ground. They apply a force to the earth equal to **ma**, where **m** is the shaker mass and **a** is its (case) acceleration. The maximum force magnitude they can apply without losing contact with the ground is therefore **mg**. The greater the shaker mass, then the greater is the force able to be applied to the earth. According to the shaker specifications (Appendix B), they are capable of producing 10 lbs of force (44.5 N) in continuous operation. No maximum force specifications for small duty cycle transient operation is given; previous experience is that ten times the specified continuous maximum is possible without damage.

A laboratory experiment was performed to quantify the shaker source output as a function of weight added to the shaker case. An accelerometer (see Section IV.A. Equipment Listing) was mounted to the top of the shaker using 5 minute epoxy. The function generator was used to input one cycle of a 100 Hz sine wave with amplitude of 1.41Vpeak to the amplifier. The output of the amplifier was used to drive the source. The source was stationary, sitting on top of the sand in the lab sand tank. A seismometer receiver was also sitting on top of the sand located 20 cm from the source, center to center. The output of the accelerometer was amplified by a gain of 10. Three signals were monitored using the oscilloscope: amplifier output, accelerometer output, and seismometer vertical output. Voltage sensitivity of the accelerometer was 10.24 mV/g. Weight was added to the top of the shaker, and the output of the amplifier was adjusted to produce 1 g of peak acceleration as indicated by the accelerometer. Table 2 shows the results as weight was added in two pound increments. A plot of the data shown in the table is also shown in Figure 21.

Weight added (lbs)	**Input Voltage (Vpk-pk)**	**Seismometer Vertical Output (mVpk-pk)**
0	14.6	56.0
2	22.2	49.0
4	28.8	54.0
6	32.4	51.0
8	37.2	59.0
10	43.2	65.0
12	57.0	85.0
14	63.4	82.0
16	67.6	95.0
18	73.4	100.0
20	80.0	109.0
22	87.6	104.0
24	100 (Clipping Negative)	122.0

Table 2. Lab Weight Testing Results

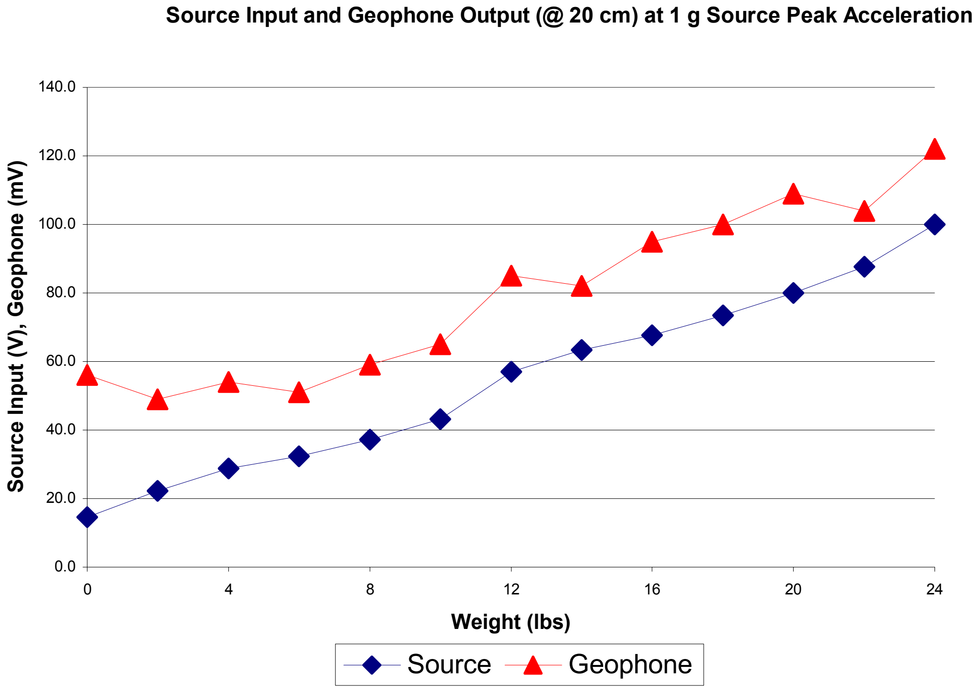


Figure 21. Lab Weight Testing Results

The test results show that the signal received at the seismometer increased approximately linearly as weight was added, for a constant 1 g of peak acceleration of the source. Once the weight increased above 22 pounds, the amplifier reached its maximum output voltage and began clipping the signal. Since the amplifier could easily provide 80 Vpk-pk for an added weight of 20 pounds, and the increase in signal output with this added weight was significant, this drive voltage was used for all field tests of the four-element end-fire array seismo-acoustic SONAR source.

D. VECTOR POLARIZATION FOR RAYLEIGH WAVES

As mentioned in the Section III, Rayleigh waves are unique in that the vertical and radial components are 90 degrees out of phase. This phase relationship causes the resulting particle motion to be elliptical. [Ref. 13] Vector polarization is a method of filtering out the unwanted body waves, which have components that are in phase.

If the phasor vertical and radial ground velocities are $\overline{V}$ and $\overline{R}$, respectively, the “vector polarization filtered” signal, or the “complex power”, is taken here to be $\overline{V}^* \overline{R}$, where $\overline{V}^*$ is the complex conjugate of $\overline{V}$. The phasors, $\overline{V}$ and $\overline{R}$, are obtained from the measured real-valued vertical and radial velocities by a Hilbert transformation operation [MATLAB],

vertical velocity signal, $\overline{V} = V e^{j(\omega t + \phi v)}$;

radial velocity signal, $\overline{R} = R e^{j(\omega t + \phi r)}$.

Since, for retrograde elliptical motion, the radial velocity phasor leads the vertical velocity phasor, the complex conjugate of the vertical velocity phasor is taken and then multiplied, data point by data point, times the complex radial velocity phasor. The $e^{j\omega t}$ term is cancelled, and we are left with the product of the magnitudes of the radial and vertical components times a phasor of the difference between the radial and vertical phases,

$\overline{V}^* = V e^{-j(\omega t + \phi v)}$;

$\overline{V}^* \overline{R} = VR e^{j(\phi r - \phi v)}$.

The imaginary part of this complex signal is simply the product of the magnitudes of the radial and vertical components multiplied by the sine of the phase difference,

$\mathrm{Im}[\overline{V}^* \overline{R}] = VR \sin(\phi r - \phi v)$.

Unwanted signals, which have components in phase, will be equal to zero, and are thus filtered out. Rayleigh Waves, which have components with a phase of 90°, will result in an imaginary part of the complex signal that is the product of the magnitudes of the radial and vertical components,

$\mathrm{Im}[\overline{V}^* \overline{R}] = VR$.

E. WAVE SPEED DETERMINATION

To determine the appropriate time delays to apply to the array elements to produce maximum output at end-fire, it is necessary to determine the wave speed in the sand on the beach. A single Aura Bass Shaker, with 20 lb of weight added, was used to generate a five cycle burst of a 100 Hz sine wave, repeated on a time interval of one second. Seismometers were placed at ranges of 1, 3, and 5 meters. The shaker and the seismometers were buried in the ground only to a depth to where the top of the device was level with the surface of the sand. To ensure both oscilloscopes were triggered simultaneously, the scopes were externally triggered from the channel A output of the delay pulse generator. All three radial signals were recorded using scope 1 and the three vertical signals were recorded using scope 2. This guaranteed accurate time delay determination between seismometers, thus accurate wave speed calculation. The recorded signals were then processed using MATLAB. First, the majority of the noise was removed by using the “smooth” function and selecting N=32. The dc component of the signals was removed by using the “detrend” function. Figures 22 and 23 show the resulting processed signals. Figure 24 shows an example of single source imaginary power at a range of 3 meters.

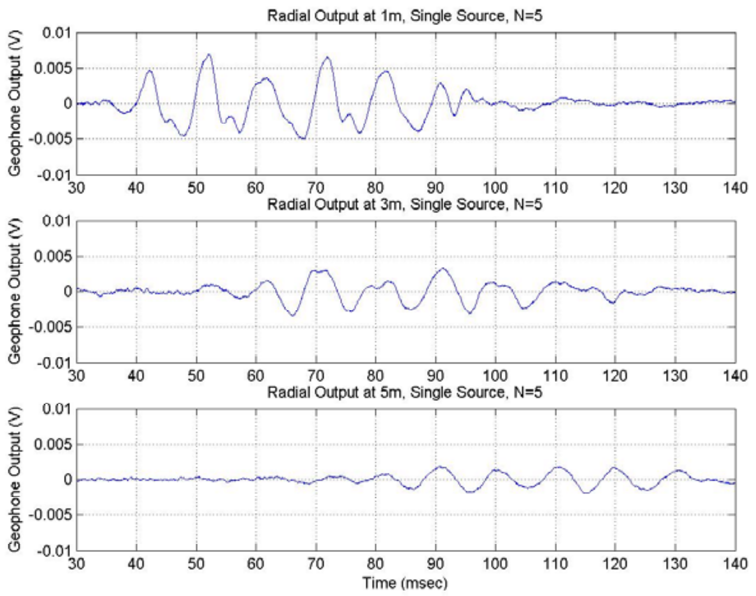


Figure 22. Single Source Radial Signals

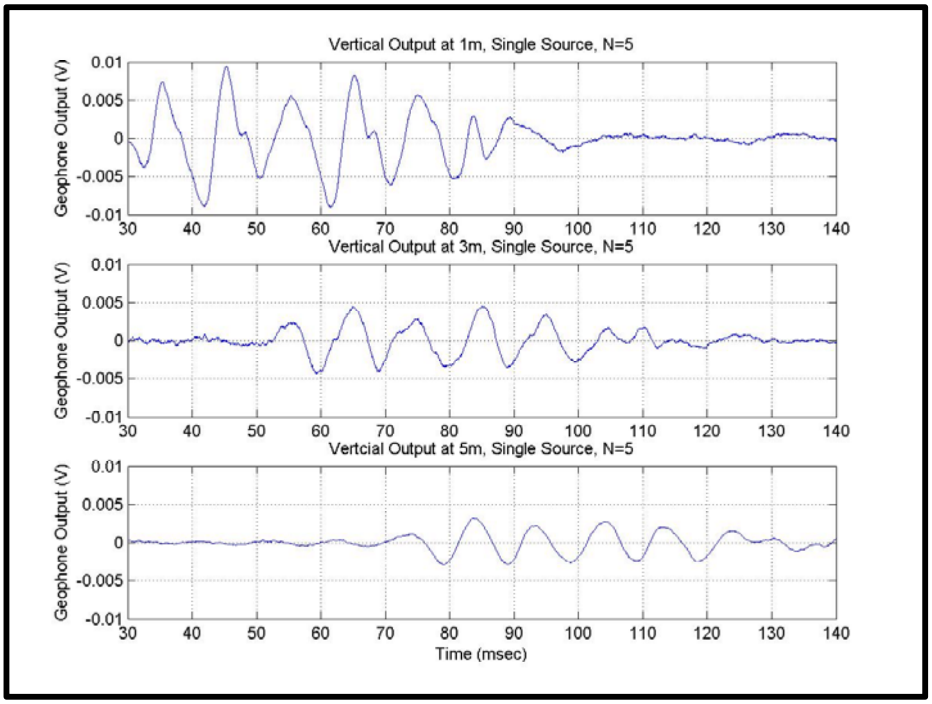


Figure 23. Single Source Vertical Signals

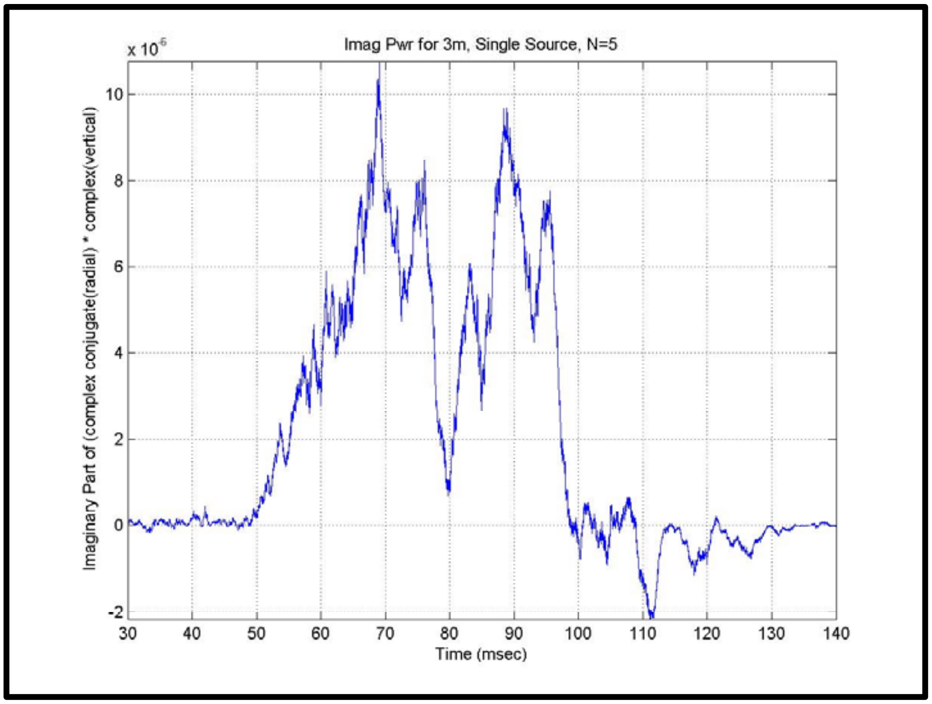


Figure 24. Single Source Imaginary Pwr at 3 meters

Next, cross-correlations of the signals were performed to determine where the maximum correlation occurred and thus the corresponding time delay and wave speed. Correlations were performed between the 1- and 3-meter signals, the 3- and 5-meter signals, and the 1- and 5-meter signals. The time delays for the maximum positive correlations were used to calculate wave speed. The individual calculated speed and the average of the three speeds are displayed on the correlation plots (Figures 25 and 26). In addition to the radial and vertical wave speeds, the imaginary power wave speed was also calculated. The phrase “imaginary power” is used here to mean the imaginary results of the process of vector polarization of the radial and vertical signals. A more complete discussion of the vector polarization is given in Section VI.D (Vector Polarization for Rayleigh Waves). The imaginary power correlation plot is shown as Figure 27. MATLAB code is included in Appendix D.

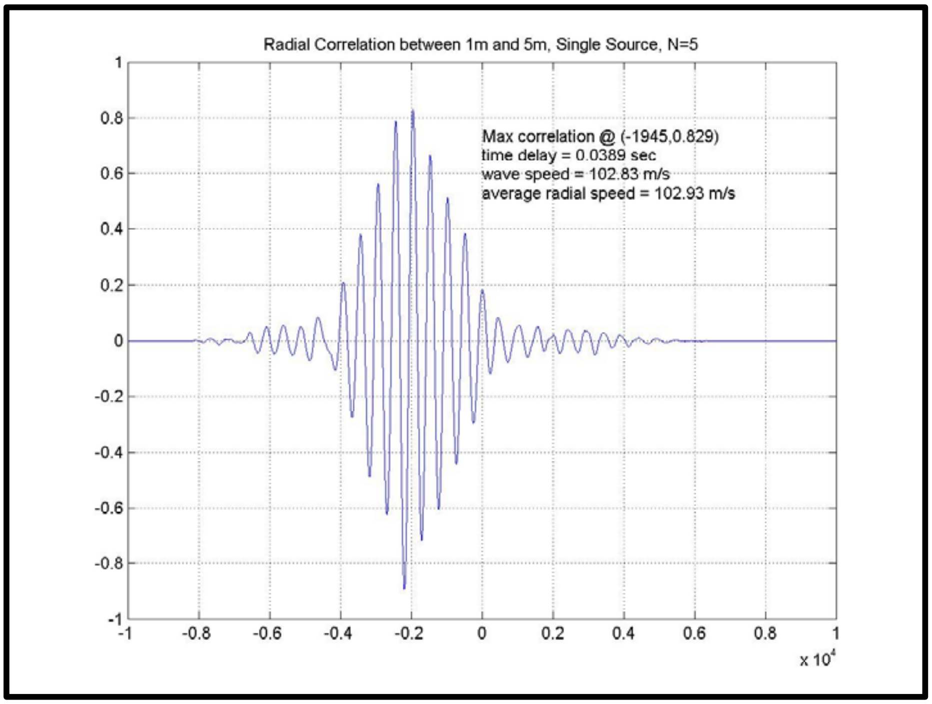


Figure 25. Single Source Radial Wave Speed Calculation

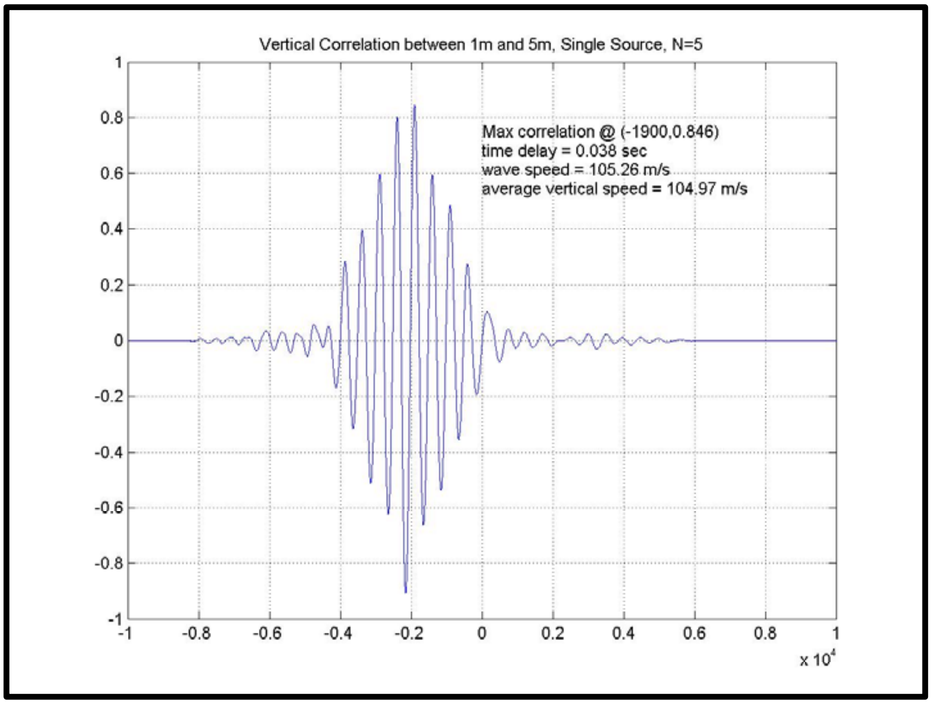


Figure 26. Single Source Vertical Wave Speed Calculation

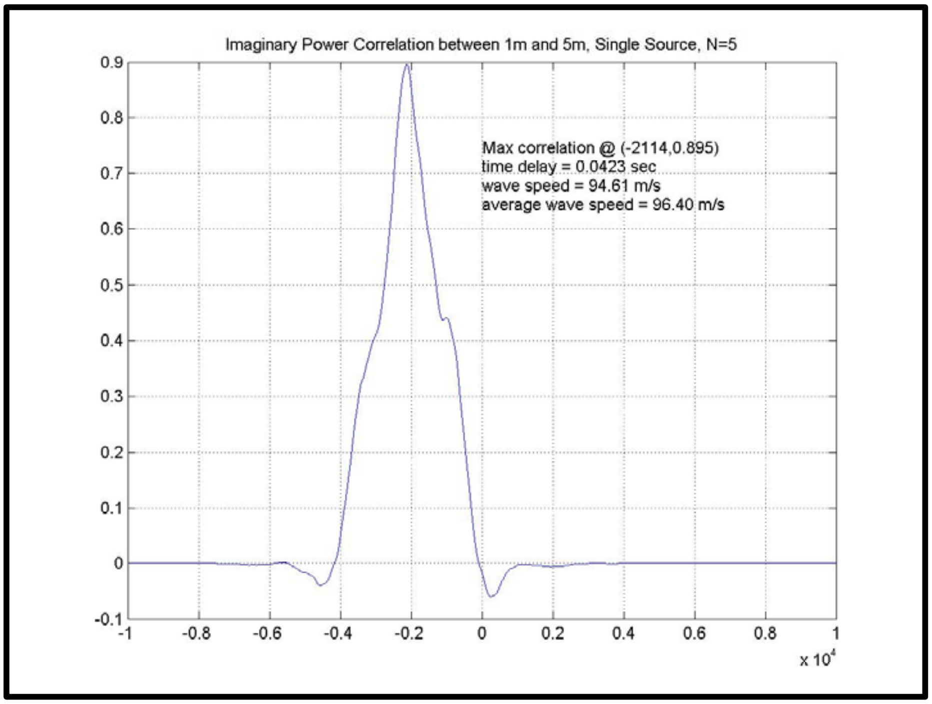


Figure 27. Single Source Imaginary Pwr Wave Speed Calculation

Results of the single-source wave speed testing were tabulated and are shown in Table 3 below. The results are consistent with previous research and show that the wave speed can be safely approximated as 100 m/s for the array design criteria, and that slower speed is usually obtained from imaginary power correlation.

Range (m)	Radial (m/s)	Vertical (m/s)	Imag Pwr (m/s)
1 – 3	**105**	**105**	**109**
3 – 5	**101**	**104**	**85**
1 – 5	**103**	**105**	**95**

Table 3. Single Source Calculated Wave Speeds

F. TIME DELAY VARIATIONS

Using four Aura Bass Shakers, each with 20 lbs of weight added, a linear array of equally spaced identical elements was formed. The shakers and the seismometers were buried in the ground only to a depth to where the top of the device was level with the surface of the sand. The intended design was to use one-quarter wavelength spacing with 90° phase difference between elements to suppress the output in the vertical downward direction, thus suppressing the unwanted P-waves. To determine the required element spacing, single-source experiments were conducted on the beach to determine wave speed (Section VI.E). Radial and vertical velocity signals were measured at various ranges, and then the signals were correlated using MATLAB cross-correlation to determine the time delays, thus the wave speeds, for measured outputs at specific ranges. The average speed measured was approximately 100 m/s. Using a frequency of 100 Hz, this gives a wavelength of 1 m. The array was chosen to consist of four elements with one-quarter wavelength spacing, so the required distance between each element was 25 cm. To produce an end-fire beam pattern for a wave traveling at 100 m/s, a time delay of approximately 0.0025 sec, or 2.5 msec, was used. Figure 28 shows a sketch of the array as designed.

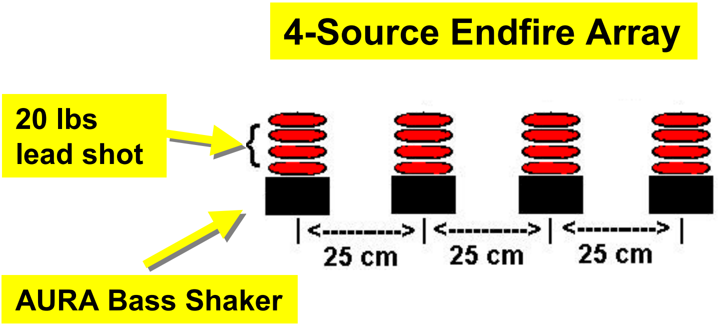


Figure 28. Four-Element End-fire Array

Testing was conducted to determine the effects of varying the time delay while keeping all other factors constant. The beach testing site and equipment setup are shown in Figures 29 and 30 below.

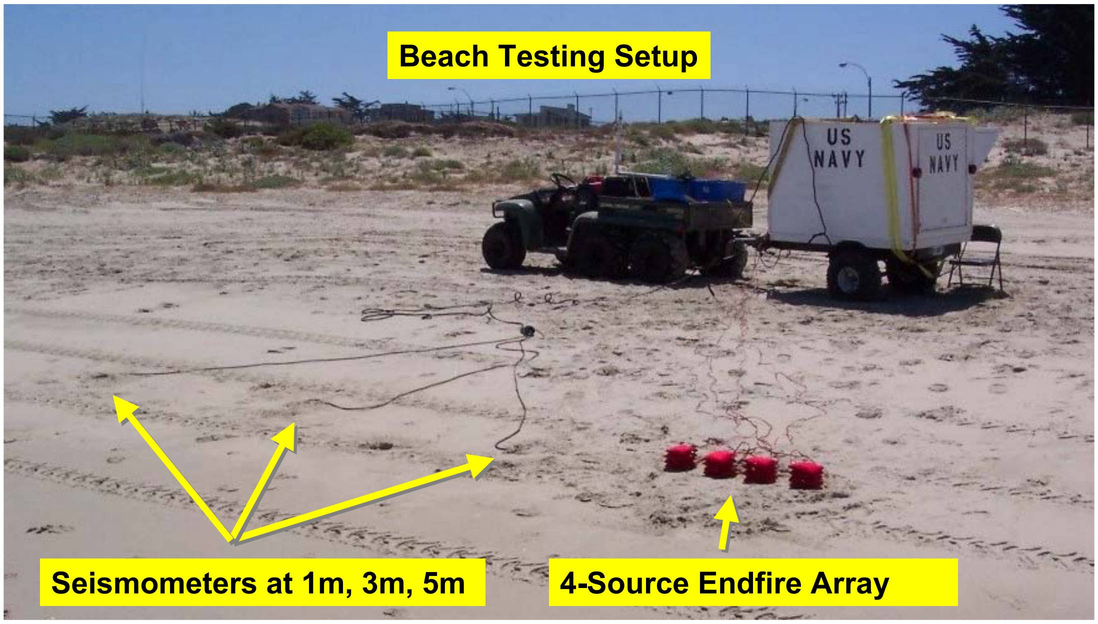


Figure 29. Beach Testing Setup

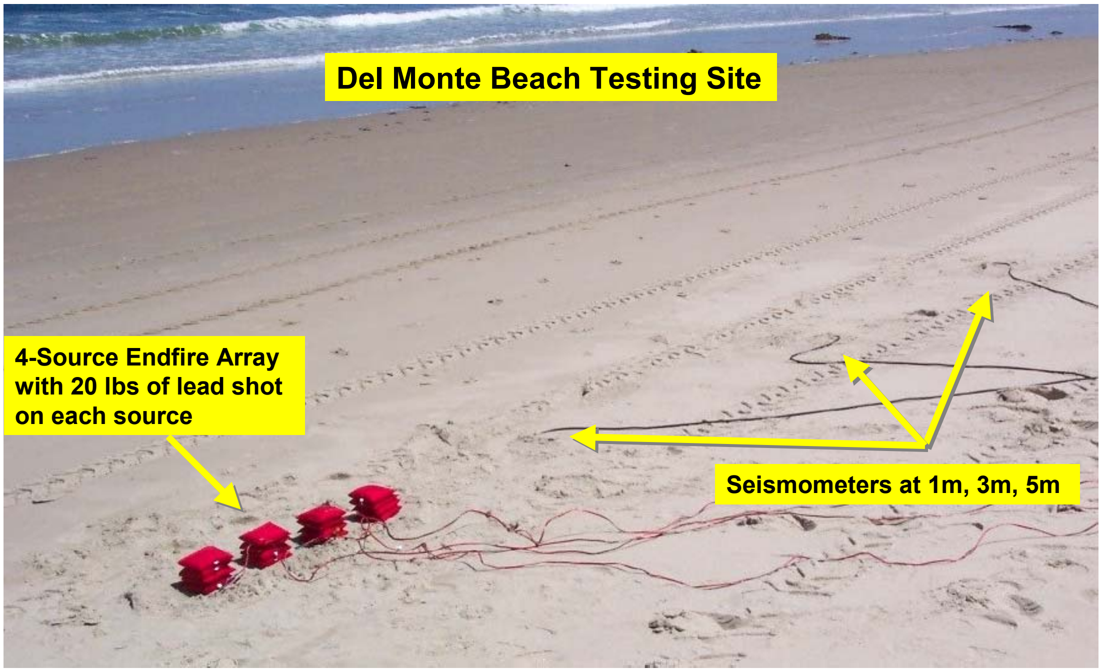


Figure 30. Del Monte Beach Testing Site

The first set of data was taken using a five-cycle burst of a 100 Hz sine wave as the source input. From the following three figures, it is noted that the different time delays affect the radial, vertical, and imaginary power signals in different ways. For the radial signal (Figure 31), a time delay of 0.002083 sec (wave speed of 120 m/s) gives the largest positive output, whereas for the vertical signal (Figure 32), a time delay of 0.002778 sec (wave speed of 90 m/s) gives the largest positive output. It should be mentioned that for the radial signal, there isn’t actually much difference in the peak signal for the different time delays. The time delay of 0.002778 sec (wave speed of 90 m/s) clearly yields the largest peak vertical signal and also the largest peak imaginary power signal (Figure 33).

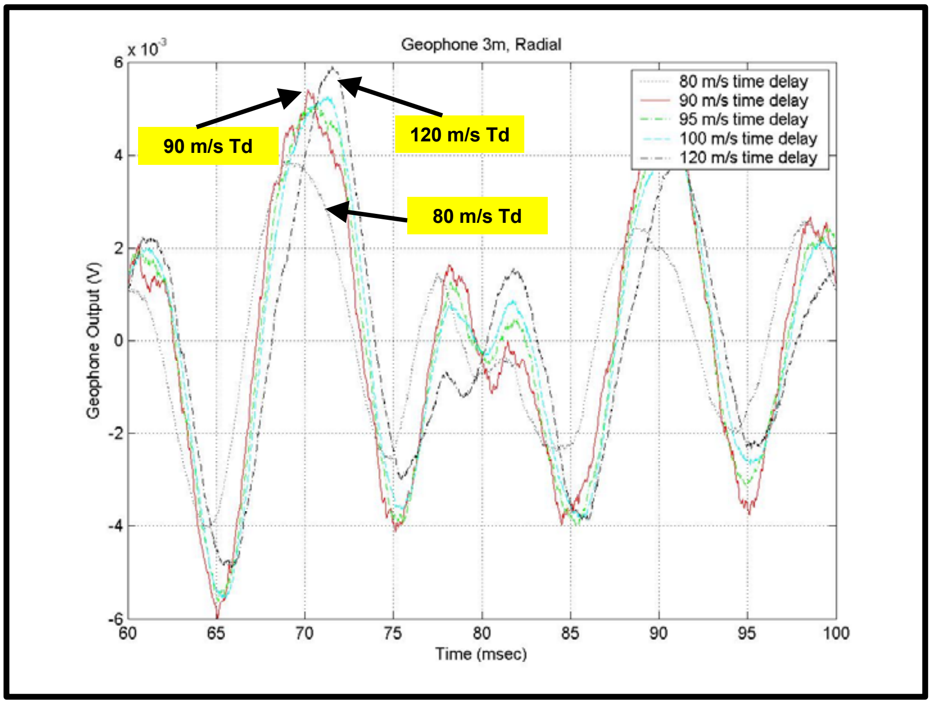


Figure 31. Vary Time Delay, 5-Cycle Burst, Radial Signal

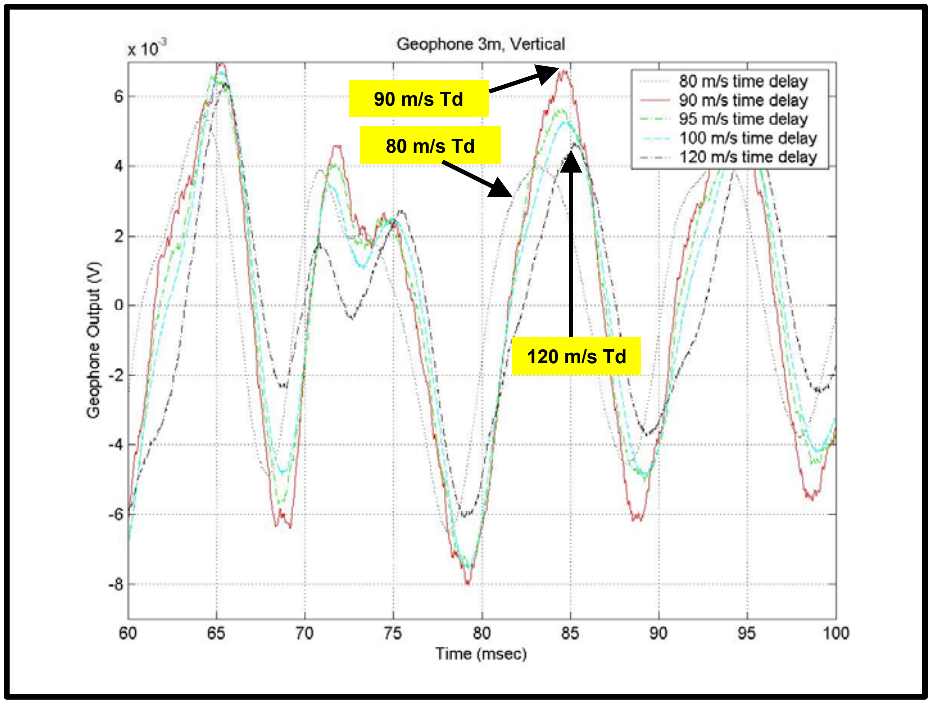


Figure 32. Vary Time Delay, 5-Cycle Burst, Vertical Signal

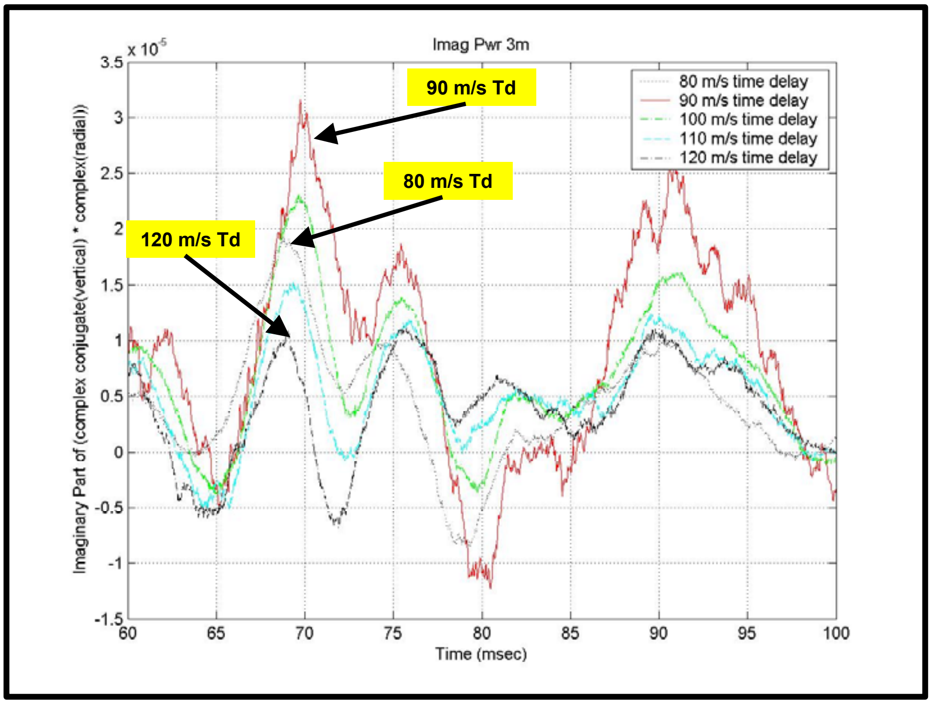


Figure 33. Vary Time Delay, 5-Cycle Burst, Imaginary Pwr Signal

The second set of data was taken using a one-cycle burst of a 100 Hz haversine wave as the source input. From the following three figures, it appears that the time delay of 0.002778 sec (wave speed of 90 m/s) gives the best overall output. This time, for the radial signal (Figure 34), a time delay of 0.002778 sec (wave speed of 90 m/s) gives the largest positive peak output, whereas for the vertical signal (Figure 35), a time delay of 0.002083 sec (wave speed of 120 m/s) gives the largest positive peak output. Again, the time delay of 0.002778 sec (wave speed of 90 m/s) yields the largest peak imaginary power signal output (Figure 36).

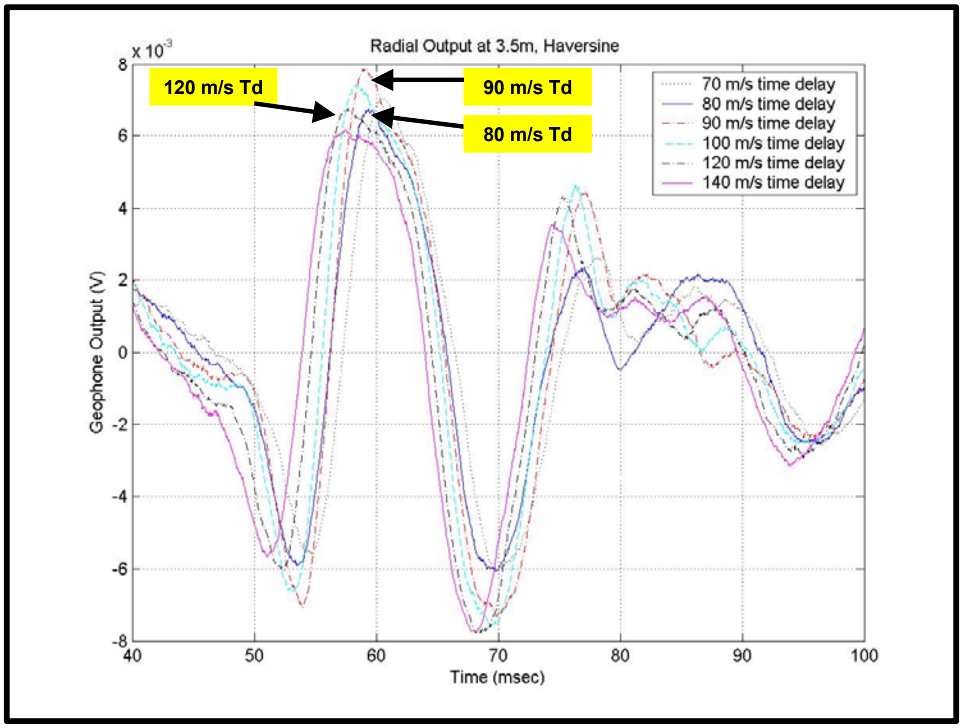


Figure 34. Vary Time Delay, Haversine, Radial Signal

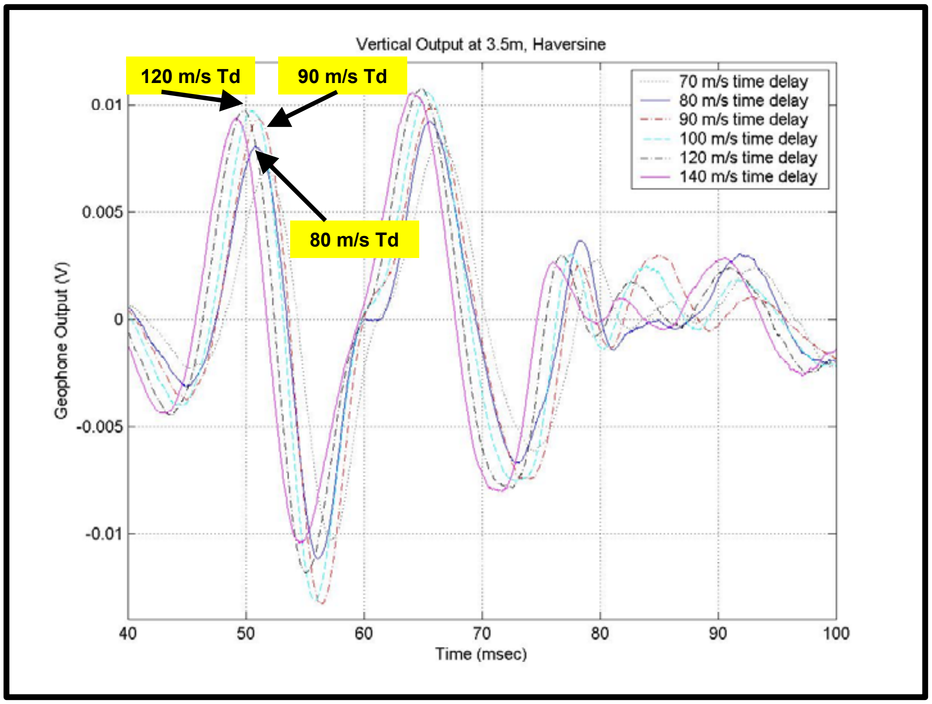


Figure 35. Vary Time Delay, Haversine, Vertical Signal

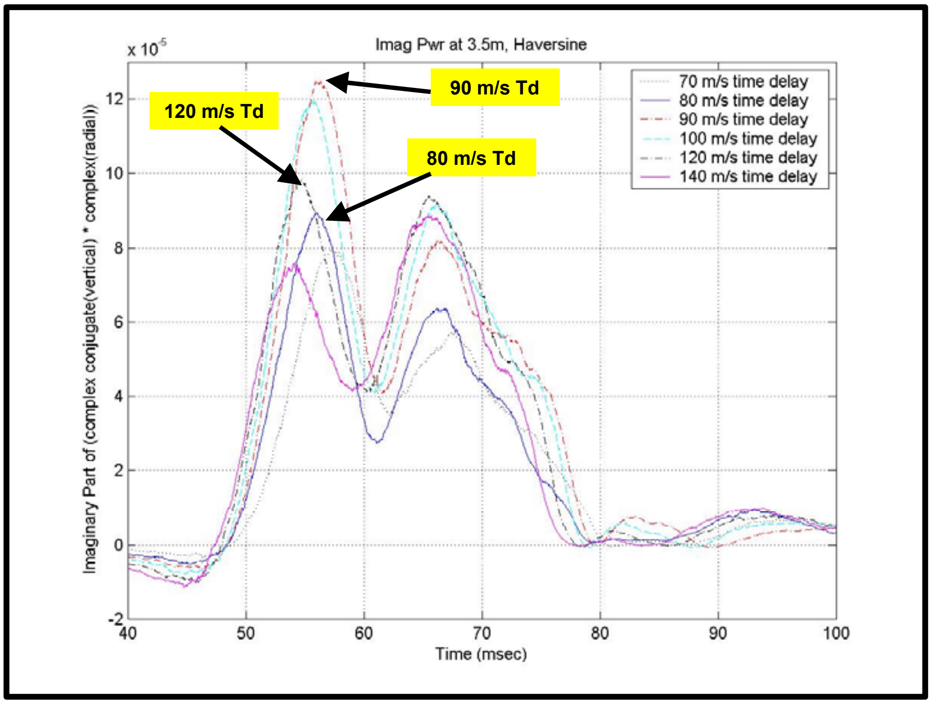


Figure 36. Vary Time Delay, Haversine, Imaginary Pwr Signal

G. RAYLEIGH WAVE CONFIRMATION

To confirm that the array preferentially excited Raleigh waves, both the radial and the vertical signals were recorded for each seismometer. As discussed in the Section VI.D, Rayleigh waves are easy to distinguish. The radial velocity of the wave leads the vertical velocity by 90° of phase, causing a retrograde elliptical motion as the wave propagates. By plotting the radial and vertical signal versus time, referred to as a Hankel plot, evidence of the retrograde elliptical motion propagation can confirm that the signal excited is indeed a Raleigh wave.

The first plot below (Figure 37) shows the signal produced by a single source using a five-cycle burst of a 100 Hz sine wave as the source input. There is evidence of some elliptical motion, but it is not very well pronounced. The second plot (Figure 38) shows the difference that was made using the same input signal, but this time using the entire array instead of only one source. There is definite retrograde elliptical motion. The third plot (Figure 39) shows a very nice, obvious Rayleigh wave pattern, which was the result of using a one-cycle burst of a 100 Hz haversine wave as the input to the four-element end-fire array.

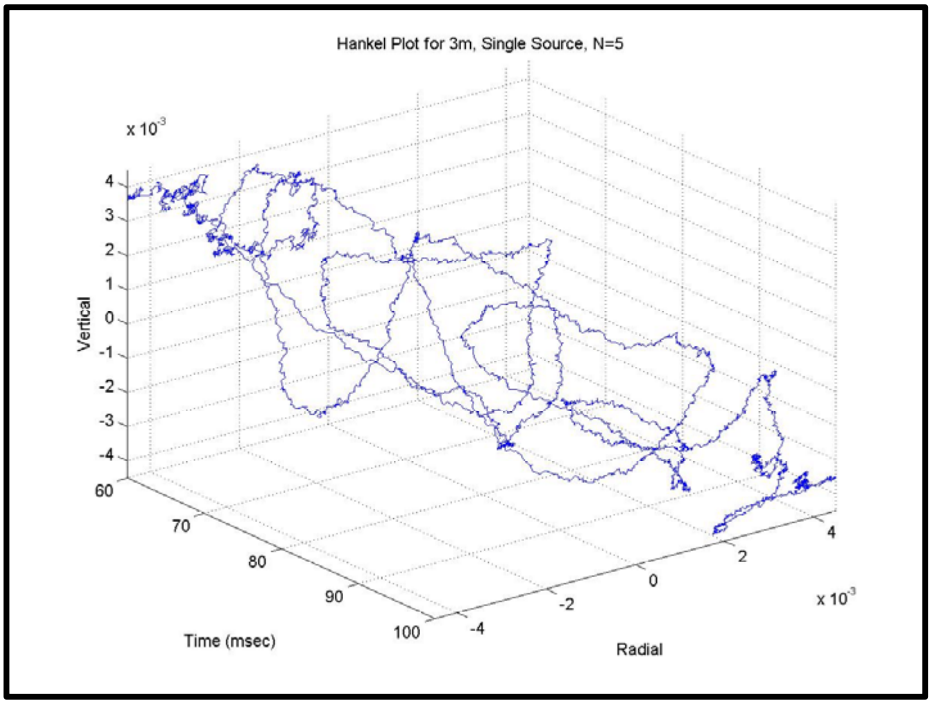


Figure 37. Hankel Plot, Single Source, 5-Cycle Burst, 3 meters

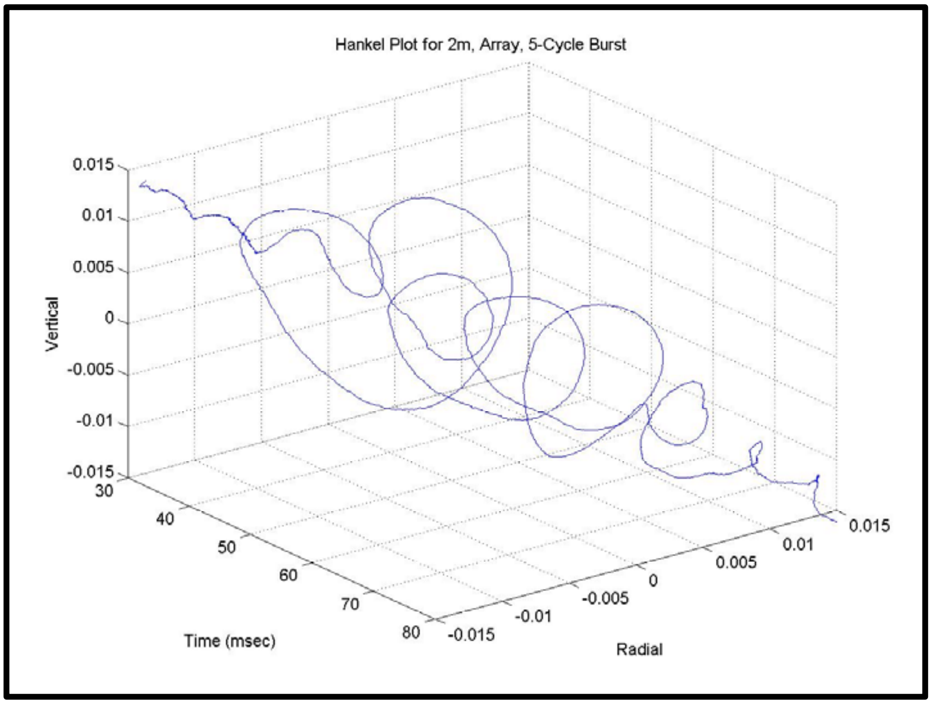


Figure 38. Hankel Plot, Array, 5-Cycle Burst, 2 meters

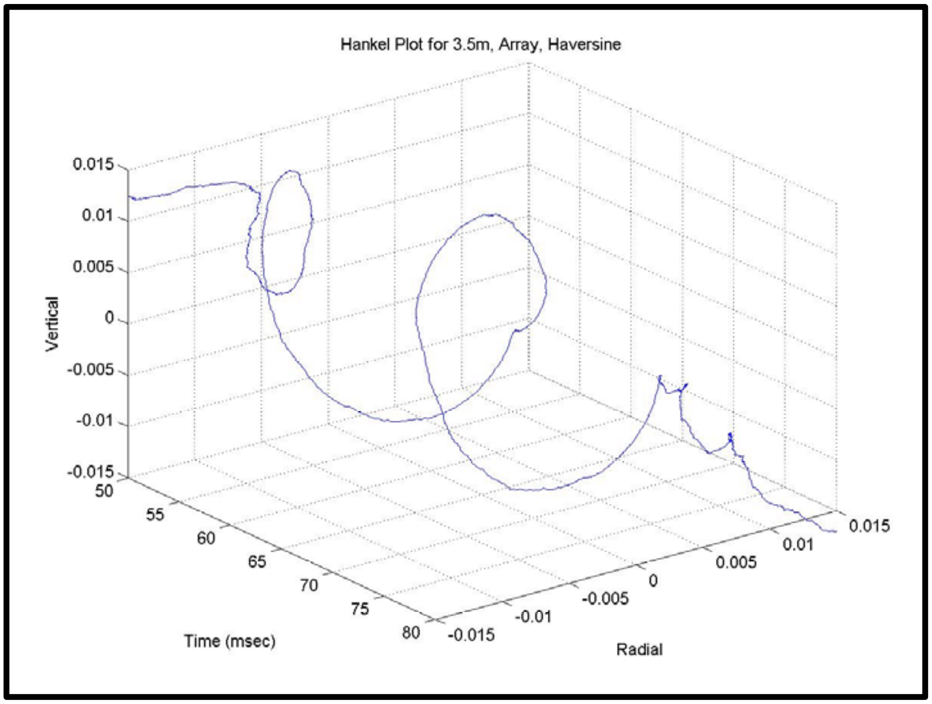


Figure 39. Hankel Plot, Array, Haversine, 3.5 meters

H. END-FIRE ARRAY WAVE SPEEDS

Vertical and radial ground velocity data were recorded to compare wave speeds using the four-element end-fire array for ranges from 1 to 17 meters. Since signal processing of the data was performed in the lab after the data were recorded, the exact wave speed for the conditions at the beach was not known at the time the experiments were performed. Based on previous testing, a time delay between elements of 0.0025 sec (wave speed of 100 m/s) was used. Again, cross-correlation was used to determine the time delay to compute wave speeds. The results are shown in Table 4. The range values indicate the location of the seismometers between which the signals were cross-correlated. As an example, 1-5 means that a cross-correlation was performed on the signal taken at 1 meter and the signal taken at 5 meters. With the exception of the calculated speed of 107 m/s (correlated between ranges 1 and 5 meters), which appears to be an outlier, the values all lie in a reasonably consistent range near 100 m/s and are comparable to the wave speeds observed for a single source.

Range (m)	**Radial (m/s)**	**Vertical (m/s)**	**Imag Pwr (m/s)**
1 – 3	**107**	**106**	**89**
3 – 5	**102**	**106**	**88**
1 – 5	**105**	**107**	**107**
7 – 9	**104**	**104**	**90**
9 – 11	**104**	**102**	**88**
7 – 11	**103**	**104**	**91**
13 – 15	**111**	**104**	**98**
15 – 17	**103**	**106**	**97**
13 – 17	**107**	**105**	**98**

Table 4. Array Calculated Wave Speeds, Td set for 100 m/s

I. END-FIRE ARRAY OUTPUT VERSUS RANGE

Ground motion data were recorded to compute the attenuation of signal strength versus range using the four-element end-fire array, as range was increased from 1.44 to 17.44 meters in increments of 2 meters as measured from the center of the array. The source input was a five-cycle burst of a 100 Hz sine wave. The time delay used was 0.0025 sec (wave speed of 100 m/s). Plots of the ground motion data were made using the peak-to-peak values for the radial, vertical, and imaginary power signals. The results are shown in Figures 40 to 42, along with theoretical curves.

Theory predicts that Rayleigh waves are expected to attenuate by a factor of $1/\sqrt{r}$ for the radial and vertical signals, and by a factor of $1/r$ for the imaginary power signal. Ground motion due to body waves radiated along the surface decay with range as $1/r^2$ ($1/r^4$ for the imaginary power). Theoretical curves are plotted which correspond to attenuation of ground motion with range as $1/\sqrt{r}$, $1/r$, and $1/r^2$ ($1/r$, $1/r^2$, and $1/r^4$ for imaginary power). It should be noted that there was no curve fitting applied; the theoretical curves were normalized to agree with the measured data at the closest range. The plots are shown with log axes for ease of comparing the constant slope theoretical values.

Figure 40 shows that the radial signal tends to follow the $1/\sqrt{r}$ curve nearer to the array but diverges slightly as range is increased. Figure 41 shows that the vertical signal follows the $1/r$ curve at closer ranges, but then tends toward the $1/\sqrt{r}$ curve as range increases. Figure 42 shows that the imaginary power tends to lie between the $1/r$ and $1/r^2$ curves. Since the recorded data are far from the $1/r^2$ (radial and vertical signals) and $1/r^4$ (imaginary power) curves, these plots show that the array definitely did not produce a body wave which traveled along the surface. To further resolve the nature of the propagating wave, more sophisticated signal processing, such as wave number filtering, could be employed.

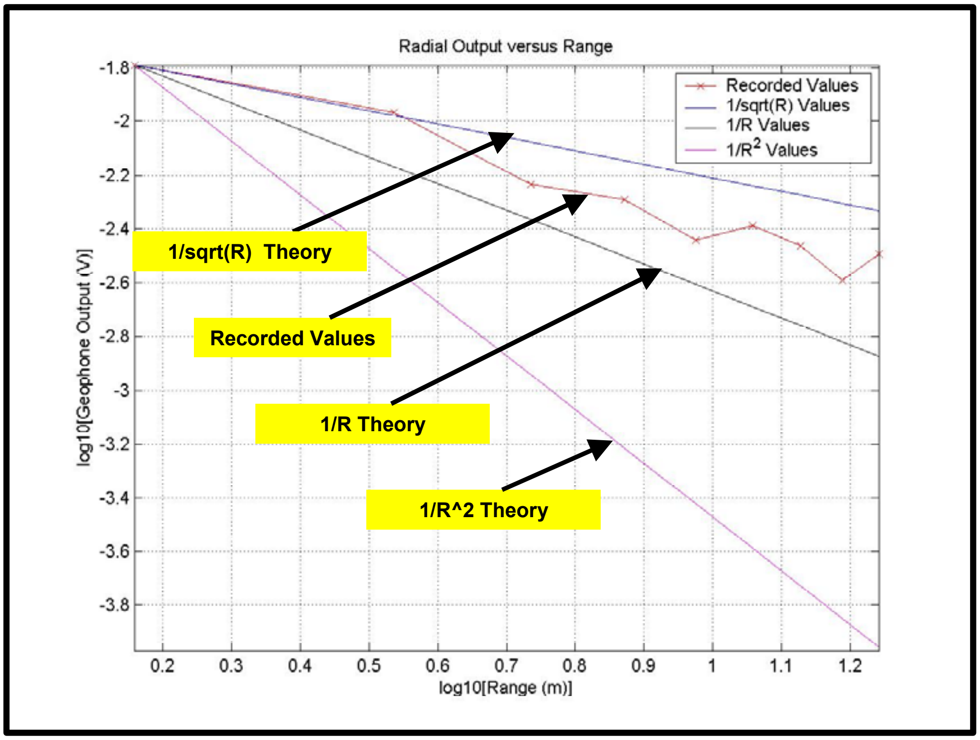


Figure 40. Array 5-Cycle Burst Radial Output Versus Range

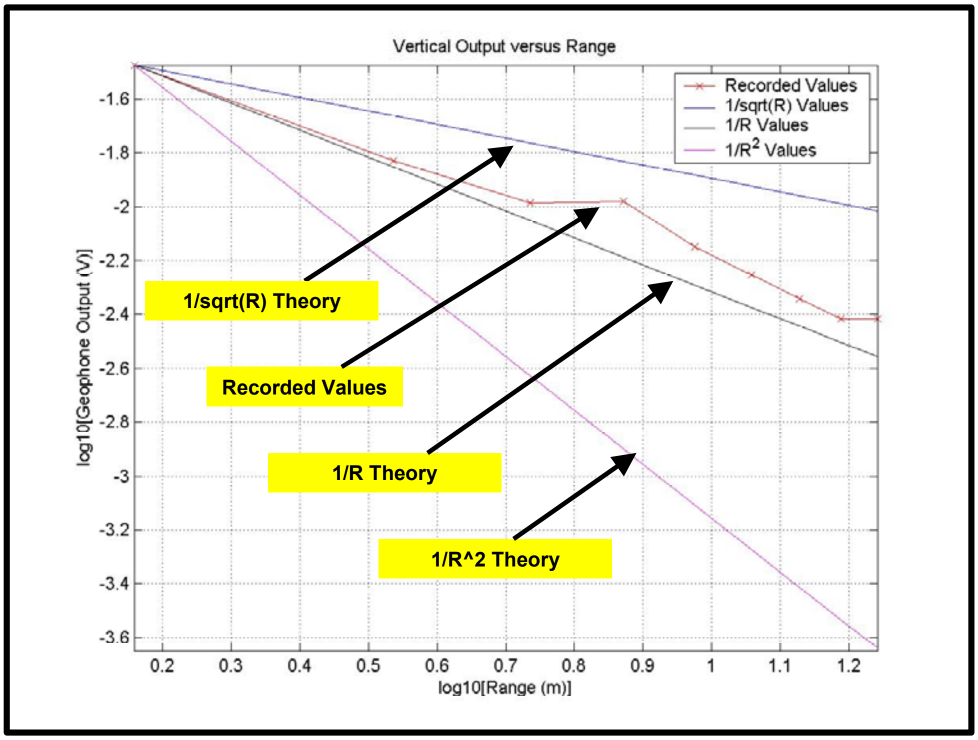


Figure 41. Array 5-Cycle Burst Vertical Output Versus Range

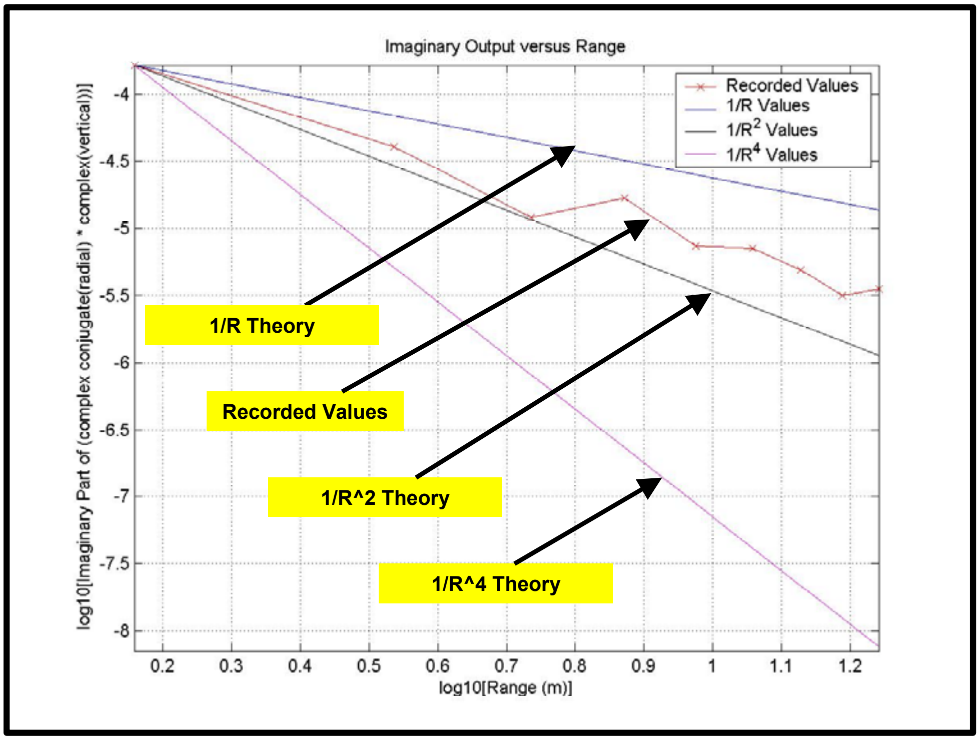


Figure 42. Array 5-Cycle Burst Imaginary Pwr Output Versus Range

J. END-FIRE ARRAY BEAM PATTERNS

Figure 43 below shows the source and seismometer layout used to conduct beam pattern measurements. Beam pattern measurements were performed with a bearing resolution of 15°. The seismometers are numbered and were positioned six at a time in the manner shown below.

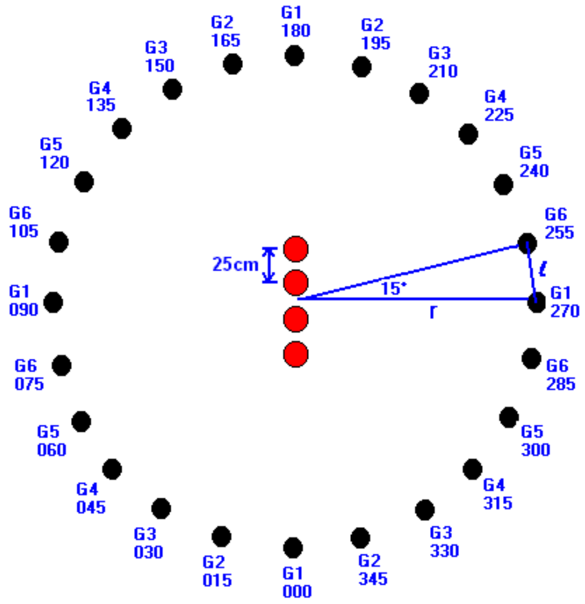


Figure 43. Beam Pattern Measurement Setup

Methodology for determining accurate seismometer locations was as follows:

Circumference of circle = $2\pi r$

15° bearing resolution gives 24 increments for complete 360°

length of arc = $2\pi r/24 = 0.261799\ r$

length of chord, ℓ, can be calculated by using simple trigonometry where

$\ell = r \sin(15°) / \sin(82.5°) = 0.261052\ r$

Calculation of the chord facilitates accurate seismometer emplacement. As an example, for $r = 3.5$m, $\ell = 91.4$cm. This allows easy use of the physics string and a meter stick (discussed in Appendix C). The seismometers are numbered 1 through 6, which ensured consistent measurements and allowed the use of calibration values.

The first beam pattern was recorded with the array positioned parallel to the water. The source was driven with a five-cycle burst of a 100 Hz sine wave repeated on a time interval of one second. Since signal processing of the data was performed in the lab after the data were recorded, the exact wave speed for the conditions at the beach was not known. Based on the previous day's testing, the time delay used was .002381 sec (for 105 m/s wave speed). The radius used was 2 meters and the bearing resolution was 15°. The distance from the source array to the edge of the water line varied throughout the measurement, beginning with approximately 32 meters and ending with approximately 3 meters. The results are shown in Figures 44, 45, and 46. The beam patterns are shown with their corresponding theoretical beam patterns for comparison by the reader. To make the plots using MATLAB, it was necessary to add 40 dB to each beam pattern value, and to clip the resulting values at zero. Consequently, the plotted scales run from 0 to 40 dB, but represent – 40 to 0 dB. For this case, the theoretical beam pattern is not optimal since the time delay is set for 105 m/s, but the element spacing is fixed at 25 cm. The recorded data show some symmetry when comparing the side adjacent to the water and the side adjacent to the sandy area of the beach. Approximately 10 dB of suppression was achieved to the rear of the array.

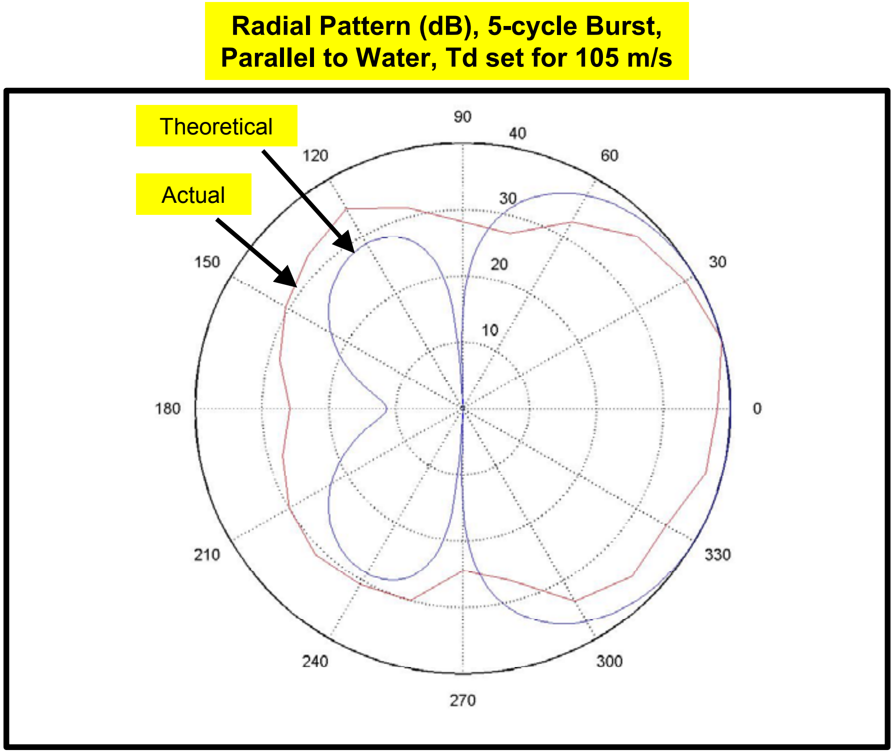

Radial Pattern (dB), 5-cycle Burst,
Parallel to Water, Td set for 105 m/s

Figure 44. Radial Beam Pattern, Radius of 2 meters

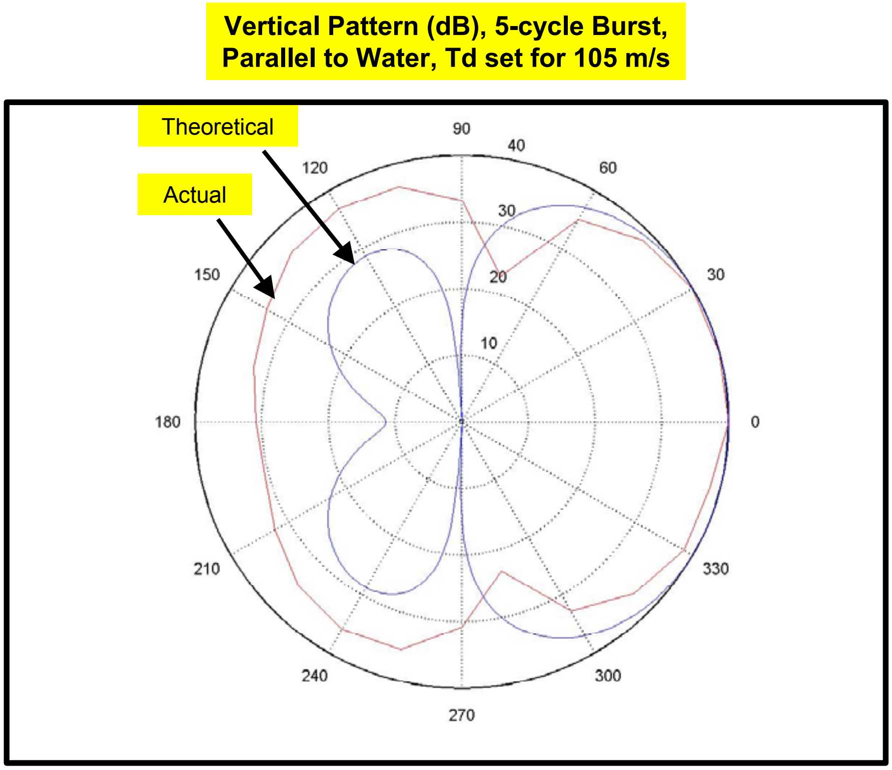


Figure 45. Vertical Beam Pattern, Radius of 2 meters

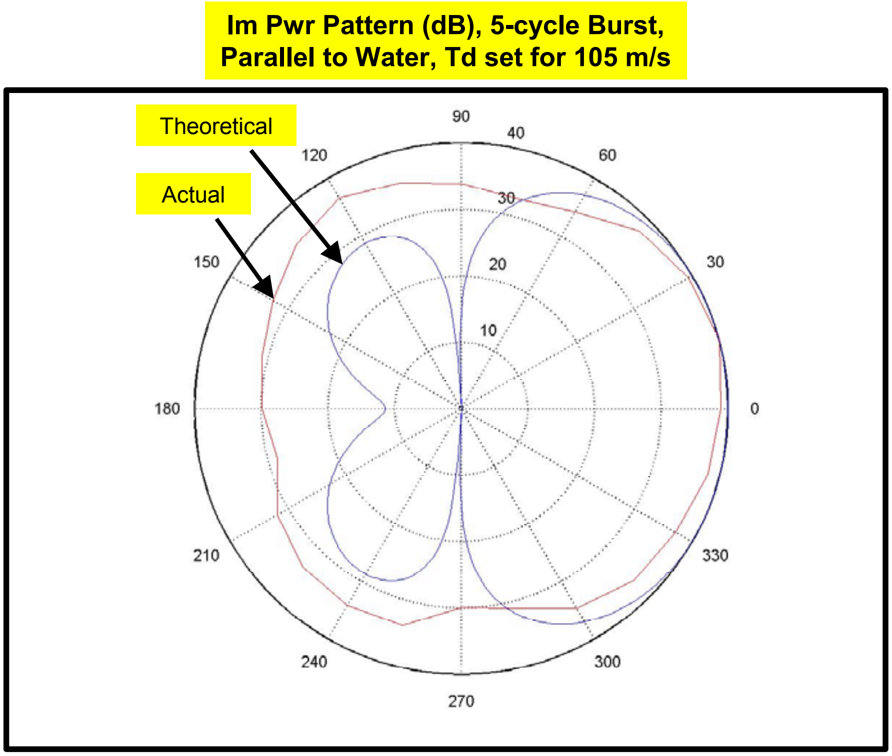


Figure 46. Imaginary Pwr Beam Pattern, Radius of 2 meters

The second beam pattern was recorded with the array positioned perpendicular to the water with the beam directed toward the beach, away from the water. The source was driven with a burst of one cycle of a 100Hz haversine wave repeated on a time interval of one second. The time delay used was 0.002381 sec (for 105 m/s wave speed). The radius used was 3.5 meters and the bearing resolution was 15°. The distance from the source array to the edge of the water line varied throughout the measurement, beginning with approximately 40 meters and ending with approximately 12 meters. The results are shown in Figures 47, 48, and 49. Again, the theoretical beam pattern is not optimal since the time delay is set for 105 m/s, but the element spacing is fixed at 25 cm. The recorded data show good symmetry with the recorded data closely following the edges of the side-lobes. Approximately 15 dB of suppression was achieved to the rear of the array.

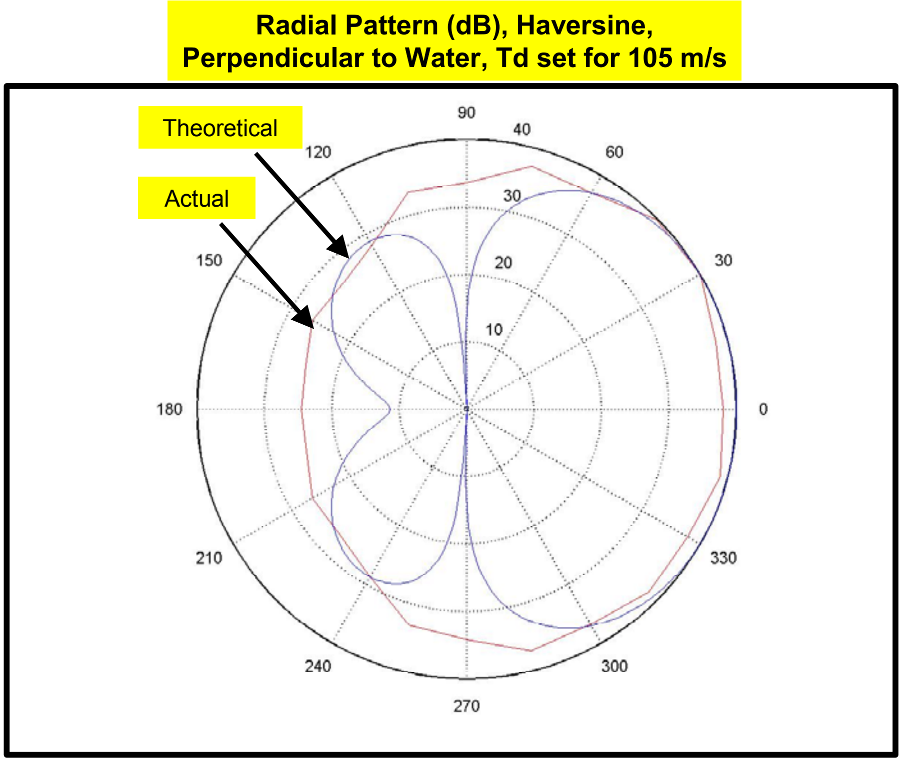
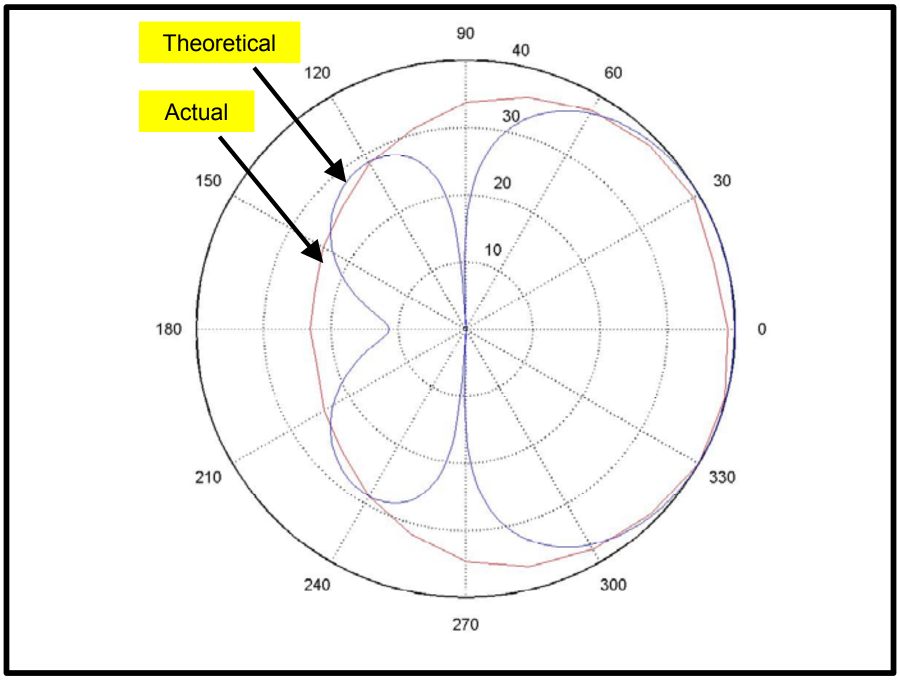
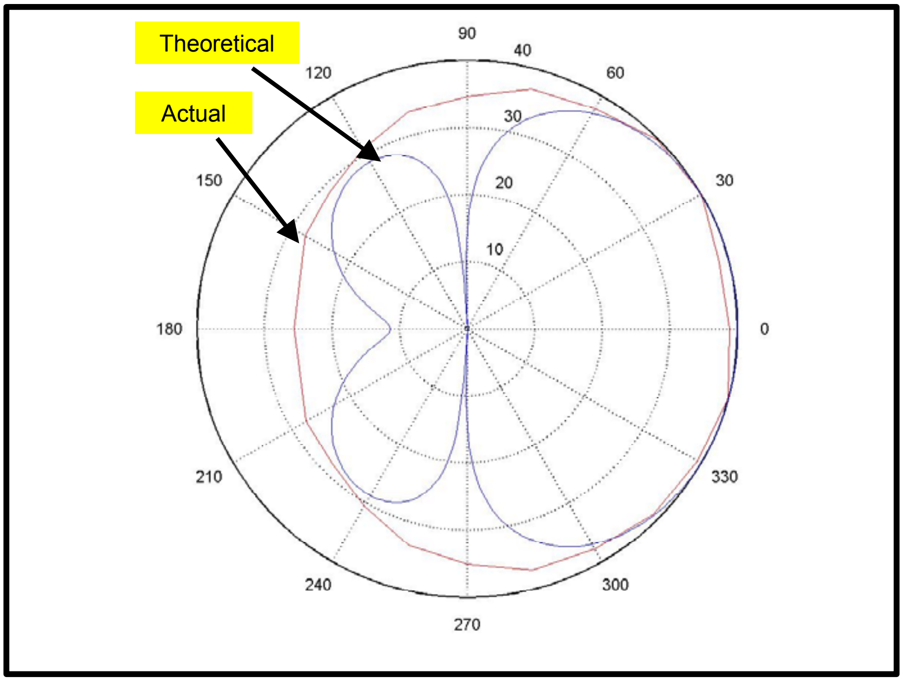


Figure 47. Radial Beam Pattern, Radius of 3.5 meters



Vertical Pattern (dB), Haversine, Perpendicular to Water, Td set for 105 m/s

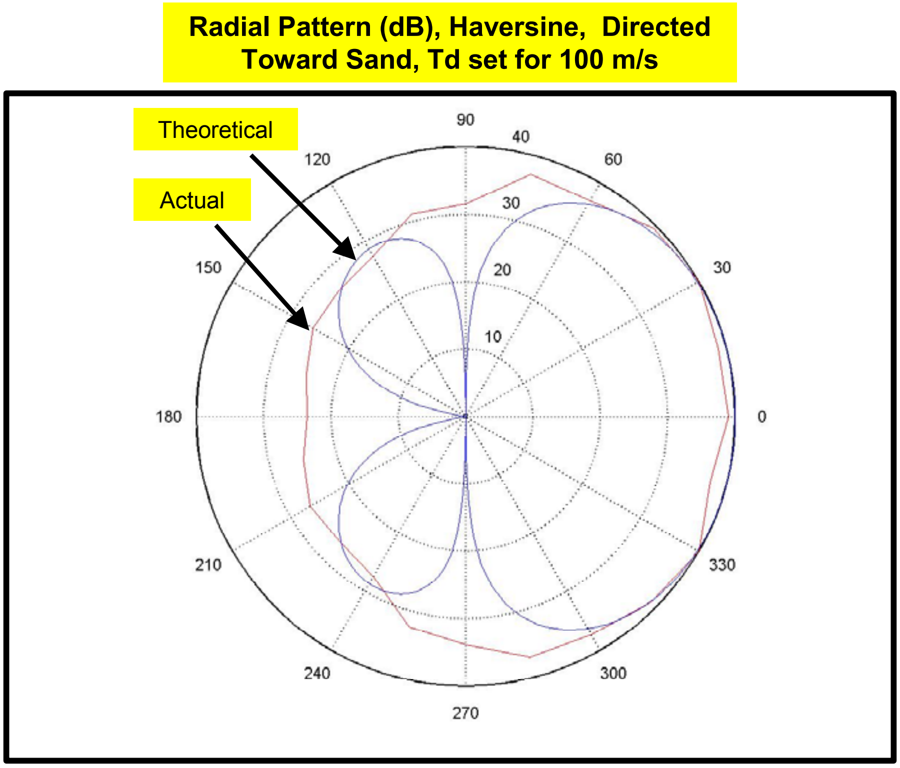
Figure 48. Vertical Beam Pattern, Radius of 3.5 meters



Im Pwr Pattern (dB), Haversine, Perpendicular to Water, Td set for 105 m/s

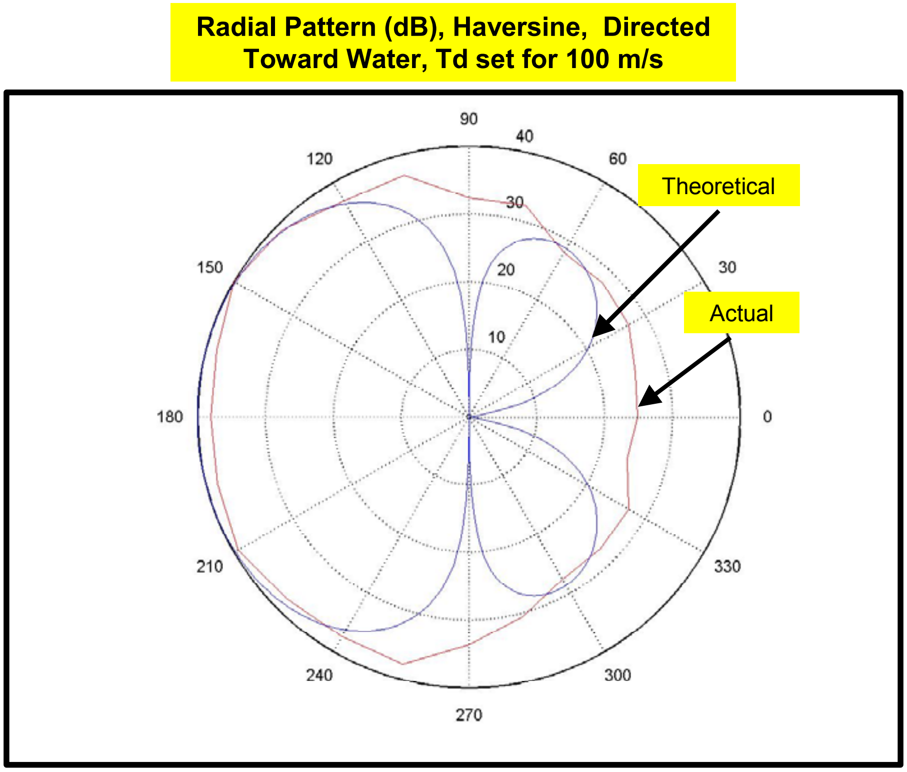
Figure 49. Imaginary Pwr Beam Pattern, Radius of 3.5 meters

A question arose as to what the effect of the water and wetter sand had on the beam pattern, or in other words, could the end-fire beam be steered equally well toward the water or toward the beach? To answer this question, the array was again positioned perpendicular to the water with the receivers placed 3.5 meters from the center of the array. The source was driven with a one-cycle burst of a 100 Hz haversine wave repeated on a time interval of one second. For this measurement, the time delay used was 0.0025 sec (for 100 m/s wave speed). This time the signals were recorded with the beam steered toward the water, and then, each time, prior to relocating the set of seismometers, the time delays were changed to steer the beam to end-fire toward the sand, and the signals were again recorded. The distance from the source array to the edge of the water line varied throughout the measurement, beginning with approximately 46 meters and ending with approximately 19 meters. The results are shown in Figures 50 through 55. The theoretical beam pattern for this case is optimal since the time delay is set for 100 m/s, and the element spacing is fixed at 25 cm. The recorded data show that there is little difference in the beam patterns when steered either toward the sand or toward the water. Again, the beam patterns have good symmetry with the recorded data closely following the edges of the side-lobes, and approximately 15 dB of suppression was achieved to the rear of the array. Table 5 shows a summary of the beam pattern results.



Radial Pattern (dB), Haversine, Directed Toward Sand, Td set for 100 m/s

Figure 50. Radial Beam Pattern, Radius of 3.5 meters



Radial Pattern (dB), Haversine, Directed Toward Water, Td set for 100 m/s

Figure 51. Radial Beam Pattern, Radius of 3.5 meters

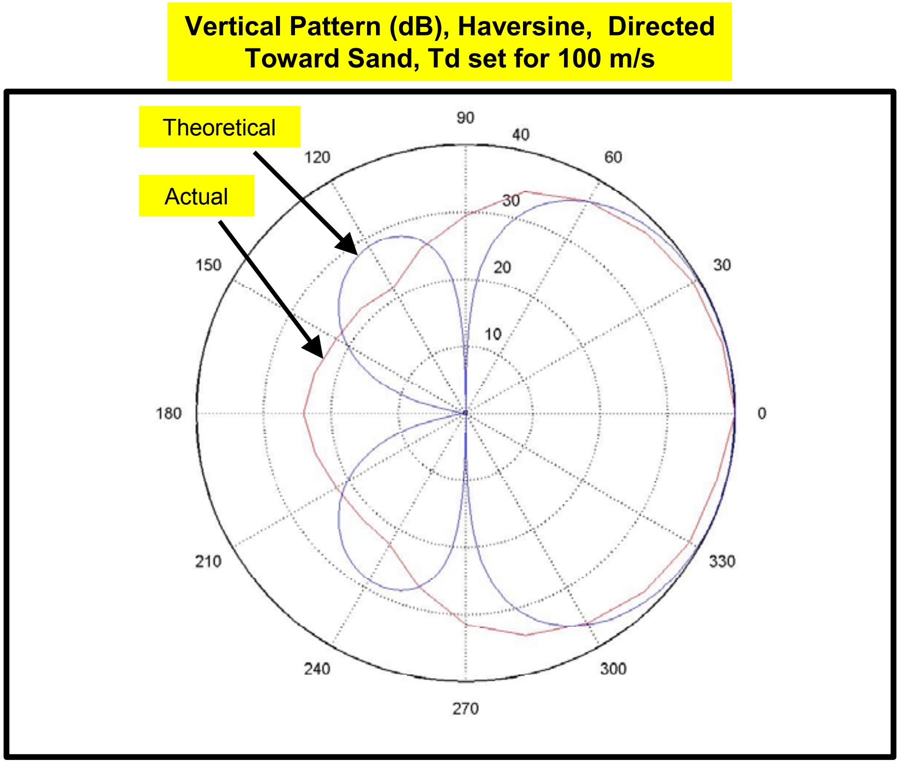


Figure 52. Vertical Beam Pattern, Radius of 3.5 meters

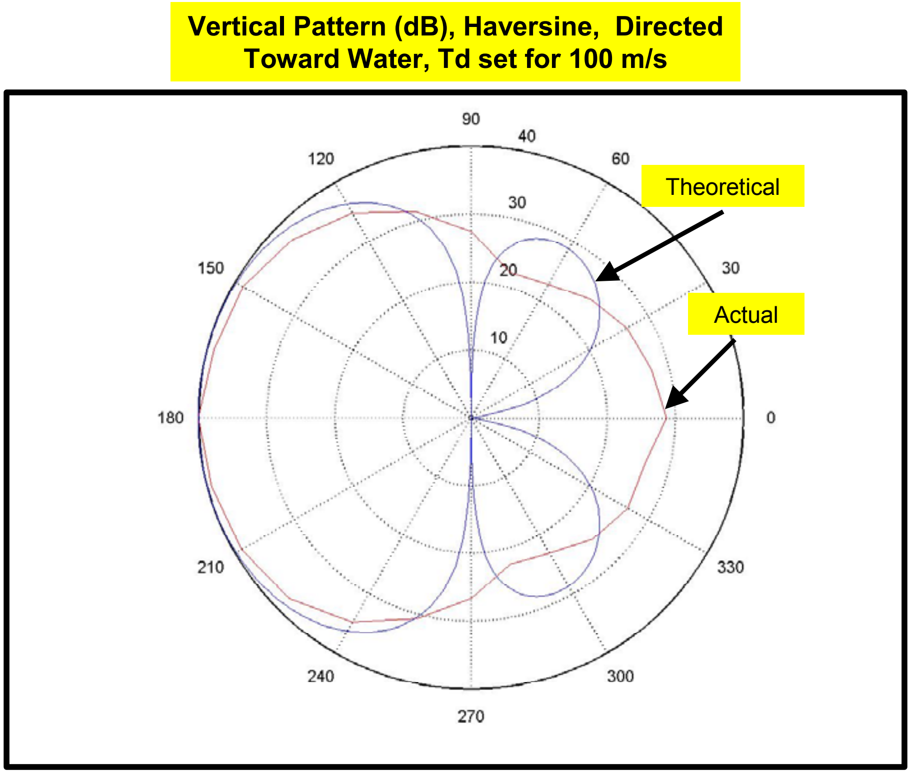


Figure 53. Vertical Beam Pattern, Radius of 3.5 meters

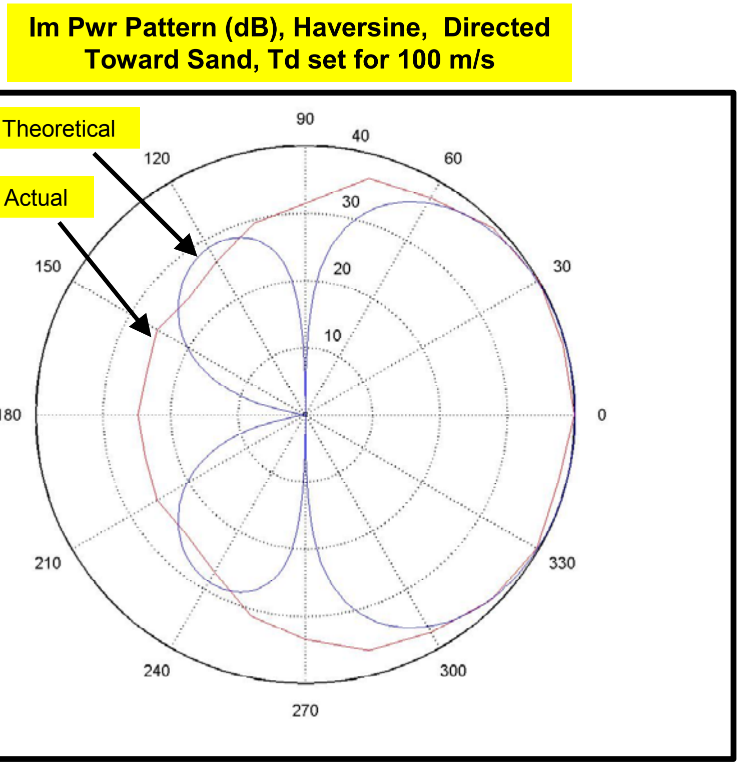


Figure 54. Imaginary Pwr Beam Pattern, Radius of 3.5 meters

z

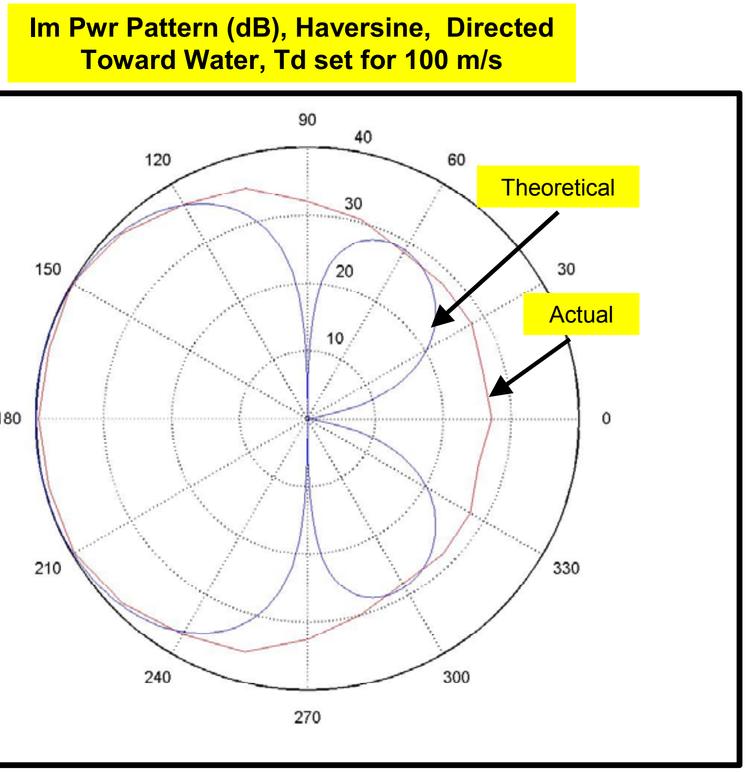


Figure 55. Imaginary Pwr Beam Pattern, Radius of 3.5 meters

Date	Array Direction	Distance to Surf (m)	Signal (100Hz Carrier)	Time Delay (in terms of wave speed)	Front to Back Suppression
7/22/03	Parallel to water-line	32 to 3	5-cycle sine wave	105 m/s	Radial: 14 dB Vertical: 9 dB Imag Pwr: 10 dB
8/8/03	Perpendicular to water-line directed toward sand	40 to 12	1-cycle haversine	105 m/s	Radial: 16 dB Vertical: 17 dB Imag Pwr: 14 dB
8/14/03	Perpendicular to water-line directed toward sand	46 to 19	1-cycle haversine	100 m/s	Radial: 16 dB Vertical: 16 dB Imag Pwr: 15 dB
8/14/03	Perpendicular to water-line directed toward water	46 to 19	1-cycle haversine	100 m/s	Radial: 16 dB Vertical: 12 dB Imag Pwr: 14 dB

Table 5. Summary of Beam Pattern Results

VII. CONCLUSIONS AND RECOMMENDATIONS

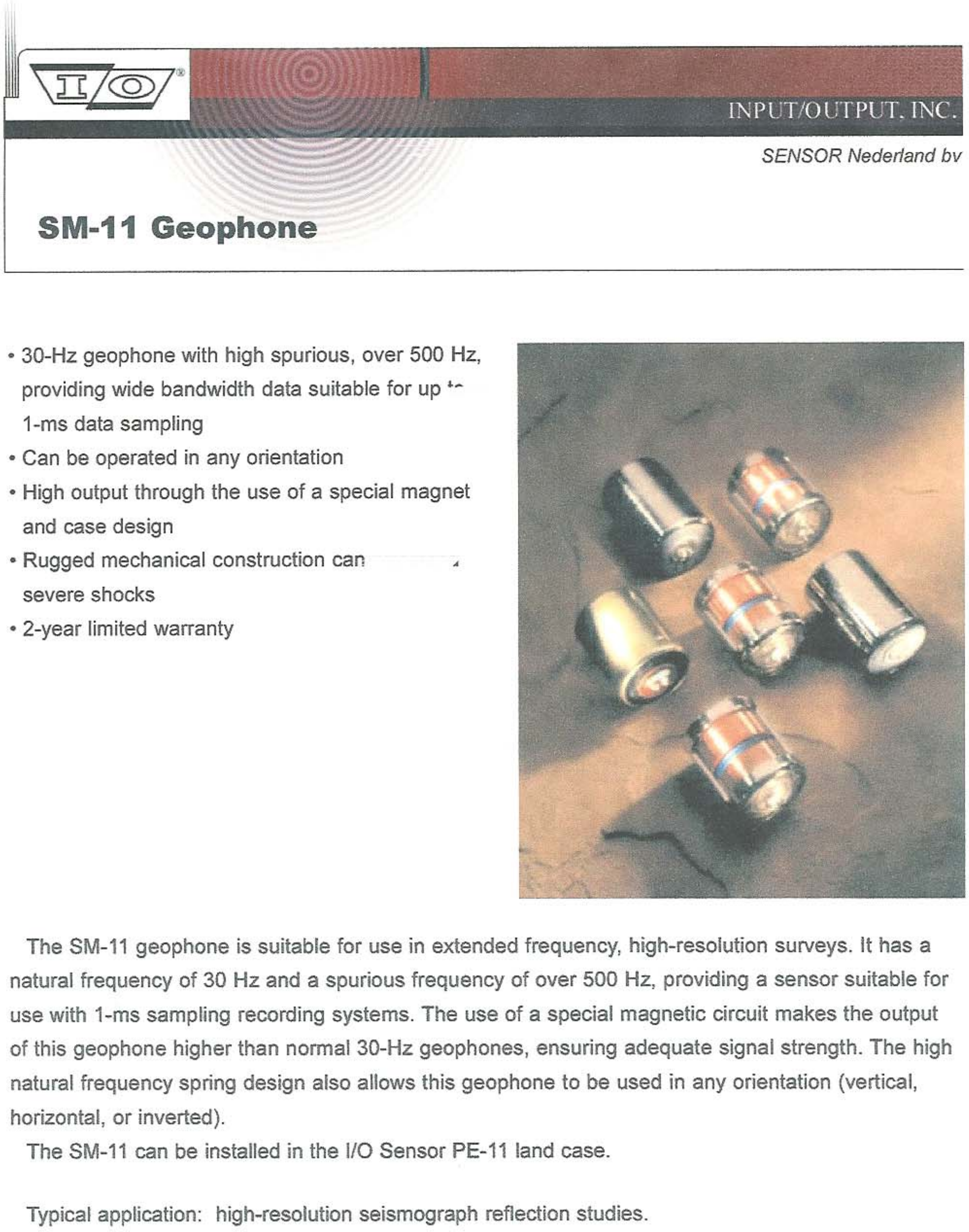
The goal of the research described in this thesis was to design and develop a four-element end-fire array to preferentially excite Rayleigh waves, to be used as a seismo-acoustic SONAR source. This goal was achieved by deploying an end-fire source array, consisting of four vertical-motion inertial mass shakers, on the sand above the surf zone at Del Monte Beach. The end-fire array was formed by applying time delays in the source drive signals to produce constructive interference in one axial direction, and destructive interference in the opposite direction. Rayleigh waves were chosen because of their unique characteristics which make them optimal for seismic mine detection, primarily because they are localized in depth near the surface. Measurements were made of the radiated vertical and radial ground motion at various ranges and angular positions for various transient signals and element-to-element time delays. These measurements were used to confirm selective Rayleigh wave production, compute the Rayleigh wave speed, determine signal attenuation versus range as compared to theory, and to make polar plots of array beam patterns to determine beam steering capability and actual versus theoretical beam patterns.

The best results were obtained when using a haversine wave as the source input. Rayleigh wave speeds were verified to be approximately 100 m/s, and the theoretical attenuation versus range for Rayleigh waves ($1/\sqrt{r}$) proved to be consistent with the recorded data. Beam steering and beam pattern results were very good; there was on average between 13dB to 15dB attenuation to the rear of the end-fire array. The results of this thesis support the use of a directional array to excite Rayleigh waves for the purpose of buried mine detection.

For future research in this area, larger amplifiers could be used to drive the sources. This would allow more weight to be added, therefore better coupling with the ground, and the increased signal strength would make target detection more probable. Also, additional elements could be added to the array to establish a narrow beam for better directionality.

Currently seismometers are being used as signal receivers. The greatest vertical motion appears directly above the buried target. It is conceivable that the ground roll vibrations due to Rayleigh waves could be sensed using a vibration-sensing laser. If this were developed, then target detection would be greatly enhanced.

APPENDIX A. SM-11 GEOPHONE SPECIFICATIONS



INPUT/OUTPUT, INC.

SENSOR Nederland bv

SM-11 Geophone

- 30-Hz geophone with high spurious, over 500 Hz, providing wide bandwidth data suitable for up to 1-ms data sampling
- Can be operated in any orientation
- High output through the use of a special magnet and case design
- Rugged mechanical construction can severe shocks
- 2-year limited warranty

The SM-11 geophone is suitable for use in extended frequency, high-resolution surveys. It has a natural frequency of 30 Hz and a spurious frequency of over 500 Hz, providing a sensor suitable for use with 1-ms sampling recording systems. The use of a special magnetic circuit makes the output of this geophone higher than normal 30-Hz geophones, ensuring adequate signal strength. The high natural frequency spring design also allows this geophone to be used in any orientation (vertical, horizontal, or inverted).

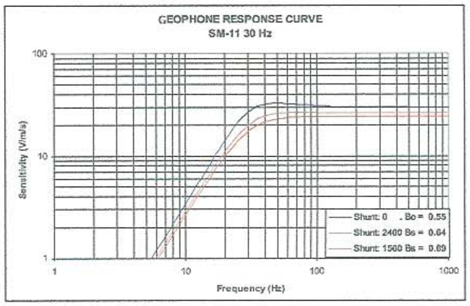
The SM-11 can be installed in the I/O Sensor PE-11 land case.

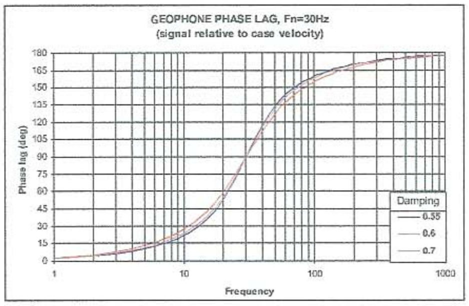
Typical application: high-resolution seismograph reflection studies.

Specifications

INPUT/OUTPUT, INC.

	SM-11/U-F
Frequency	
Natural frequency (f_n)	30 Hz
Tolerance	±5%
Maximum tilt angle for specified f_n	180°
Typical spurious frequency	>500 Hz
Distortion	
Distortion with 0.7 in/s p.p. coil-to-case velocity	<0.2%
Distortion measurement frequency	30 Hz
Maximum tilt angle for distortion specification	180°
Damping	
Open-circuit damping	0.55
Open-circuit damping tolerance	±5%
Resistance	
Standard coil resistances	360
Tolerance	±5%
Sensitivity	
Open-circuit sensitivity	30 V/m/s (0.75 V/in/s)
Tolerance	±5%
$R_tB_cf_n$	7,785 ΩHz
Moving mass	9.2 g (0.32
Maximum coil excursion p.p.	>1 mm (>0.04 in)
Physical Characteristics	
Diameter	26.6 mm (1.02 in)
Height	32 mm (1.26 in)
Weight	89 g (3.13 oz)
Operating temperature range	–40°C to +100°C (–40°F to +212°F)
Limited Warranty Period*	2 years





* Warranty excludes damage caused by high-voltage and physical damage to the element case.

All parameters are specified at +20°C in the horizontal position unless otherwise stated.

Ordering Information

SM-11

SM-11/U-FT	30 Hz	360 Ω	(upright)	P/N 1011010
SM-11/H-FT	30 Hz	360 Ω	(horizontal)	P/N 1011030

United States – Stafford, TX
Input/Output, Inc.
Fax 281.879.3500
Phone 281.933.3339

Russia
Input/Output, Inc.
Fax 7.095.2322240
Phone 7.095.2322254

England
Input/Output, Inc.
Fax 44.1603.411403
Phone 44.1603.411400

Web Site
www.i-o.com

APPENDIX B. AURA BASS SHAKER SPECIFICATIONS

Bass Shakers™ Let You Feel All the Bass without Breaking the Sound Barrier

Mfr. Sugg. Retail $199.95
WAS ~~$179.99~~ $99.99

The Bass Shaker is a transducer that generates the sensation of sound by vibration, not by moving air. The result is a big bass effect without a high pressure level that could distort sound or blow your speakers. They are also great for adding bass to vehicles with poor sound insulation, such as trucks. Mount these Bass Shakers under your front seats and you will feel the punch of the drums and the kick of the percussion, as if you were on stage or in the studio when the music was recorded! Hooks up easily to any standard audio amplifier output channel, and can be used with existing subwoofers for an extra kick. Each Bass Shaker requires only 25W RMS of power. Model AST1B4. **Item No. B-40070-492383 S/H ~~$9.99~~** *FREE*

Specifications:	AST-1B-4 Bass Shaker
Frame Size:	4.75" dia. X 2.35" h
Magnet Type:	Ceramic
Power Rating:	Nominal 25 W
Force, Nominal:	10 lbf (44.5 N)
Weight:	2.5 lbs each
Resonance Frequency (fo):	42 Hz
Usable Frequency Range:	20-100 Hz

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. BEACH TESTING AND LESSONS LEARNED

1. Tide Prediction. Equipment setup and data recording may take about four hours to perform for a complete beam pattern. Arriving at the beach at low tide usually gives a window of time of about five hours with adequate beach space for beam pattern measurements. Tide curves published by NOAA (http://co-ops.nos.noaa.gov) were very useful. This particular site allows you to select the location, dates, and to select local standard time (LST) for display. The data can be viewed either as a plot or as a series of times and water levels. Figures 22 and 23 are shown to give the reader an idea of how quickly the width of the NPS Del Monte Beach testing site can change. The data for the incoming tide prediction (predicted water level) were obtained from the NOAA website above. The data for the beach space versus time were obtained by measuring the distance from the test setup to where water was washing up on the shore at various times during a day of testing.

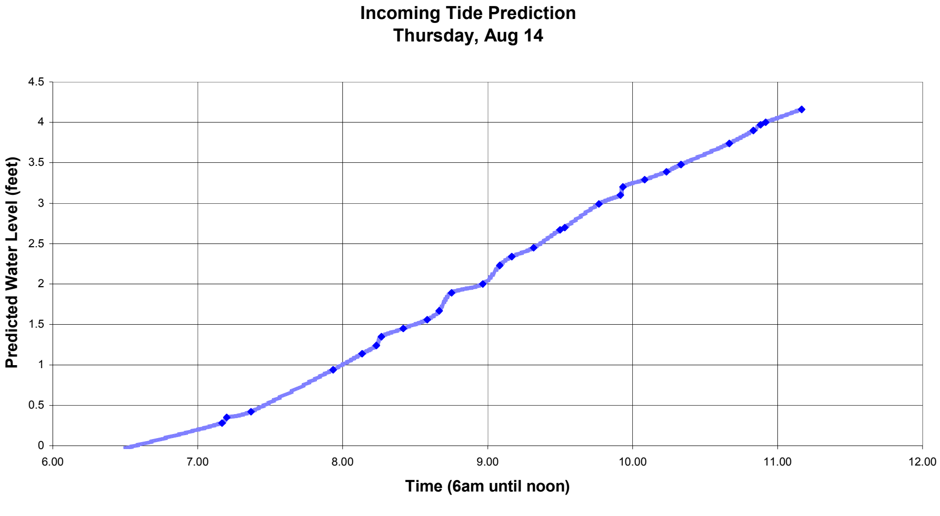


Figure 56. Incoming Tide Prediction

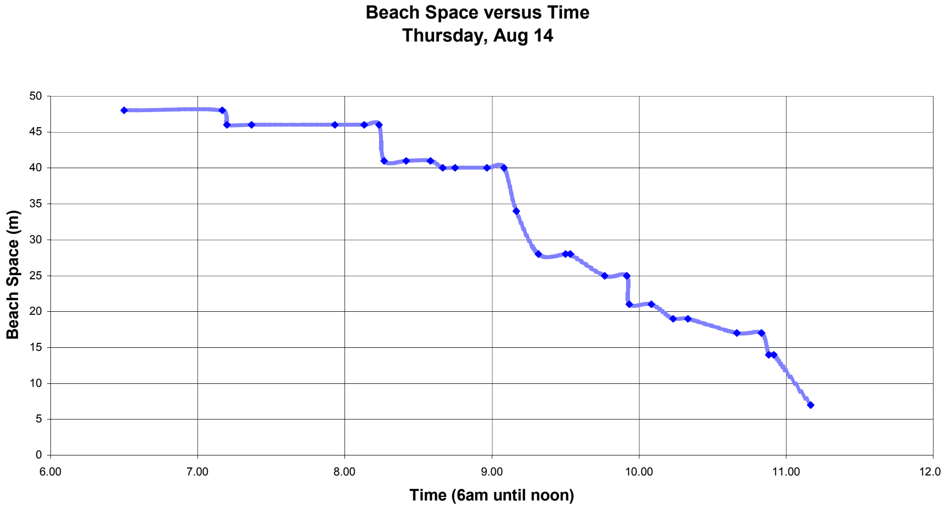


Figure 57. Beach Space Versus Time

2. Power Generator. The Honda will easily supply power for a full day of testing on a tank of gas, but one tank of gas is usually not enough for two days, so be smart and top of the Honda every time you plan to do testing.

3. Adequate Number Data Storage Disks. One 1.4 Mb disk will hold six recorded signals. Both radial and vertical components of the signals should be recorded, so each seismometer position means two recorded signals. Therefore, two disks are required for each set of six seismometer positions.

4. Equipment Connections. Leaving the majority of the equipment connected in the trailer between testing days saves a lot of time. Also, maintaining connections to oscilloscopes during testing can facilitate measurements. Using two Tektronics, 4-input, scopes, you have three useable channels since one of the channels is reserved for a trigger input (recommend channel 4). The radial signals may be on one scope and the vertical signals on the other scope. Once seismometer s 1, 2, and 3 are recorded, simply swap the BNC connections on the receiver panel to seismometer s 4, 5, and 6. Note that for Hankel plots, you will want both the radial and vertical signals on the same oscilloscope.

5. Seismometer Spacing. This mainly refers to the method of spacing seismometers for a beam pattern measurement. From various methods tried, the following works the best, especially for a one-man show at the beach: Use physics string (it doesn't stretch), with a loop tied in each end, measured to the desired radius. Have a ground rod or other device for securing physics string at center of beam pattern and a meter stick, or other measuring device, through the other physics string loop for easy determination of proper spacing between seismometers.

6. Seismometer Position. Note that the positive radial position is either pointing toward the inscribed X or Y on the seismometer, or away from the source.

7. Saving Data. One method that worked really well was to save files as spreadsheet type [.csv] and then import directly into MATLAB using the MATLAB import wizard.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. MATLAB CODE

A.   WAVE SPEED DETERMINATION

```
%  This program was written specifically for the data collected on
%  July 16, 2003.  For other data, the load command and variable names must be changed.
%  This program uses the single source data to calculate wave speed in two ways:
%  The first method uses a correlation of the recorded geophone ouputs at specified ranges to
calculate the
%  time delay and then uses the known distance to determine the speed.
%  The second method uses a correlation of the imaginary power derived from the recorded
geophone ouputs at
%  specified ranges to calculate the time delay and then uses the known distance to determine
the speed.
%  This program also uses the radial and vertical data to produce Hankel Plots.
%
clear all

load july16     % july16 contains all the data recorded at the beach on July 16

% Using 32 pt smoothing, most of the noise is removed from the signal
% "detrend" levels signal to 0

geo_1r=detrend(smooth(G1_R1(:,2),32));
geo_2r=detrend(smooth(G2_R3(:,2),32));
geo_3r=detrend(smooth(G3_R5(:,2),32));
geo_1v=detrend(smooth(G1_V1(:,2),32));
geo_2v=detrend(smooth(G2_V3(:,2),32));
geo_3v=detrend(smooth(G3_V5(:,2),32));
timeindex = [1:length(geo_1r)];
time = timeindex*.02;

% Plots the smoothed signals

figure(1)
subplot(3,1,1)
plot(time,geo_1r),  title('Radial Output at 1m, Single Source, N=5'),grid, axis([30 140 -.01 .01])
ylabel('Geophone Output (V)')
subplot(3,1,2)
plot(time,geo_2r),  title('Radial Output at 3m, Single Source, N=5'),grid, axis([30 140 -.01 .01])
ylabel('Geophone Output (V)')
subplot(3,1,3)
plot(time,geo_3r),  title('Radial Output at 5m, Single Source, N=5'),grid, axis([30 140 -.01 .01])
xlabel('Time (msec)'), ylabel('Geophone Output (V)')

figure(2)
subplot(3,1,1)
plot(time,geo_1v),  title('Vertical Output at 1m, Single Source, N=5'),grid, axis([30 140 -.01 .01])
ylabel('Geophone Output (V)')
subplot(3,1,2)
plot(time,geo_2v),  title('Vertical Output at 3m, Single Source, N=5'),grid, axis([30 140 -.01 .01])
ylabel('Geophone Output (V)')
subplot(3,1,3)
plot(time,geo_3v),  title('Vertcial Output at 5m, Single Source, N=5'),grid, axis([30 140 -.01 .01])
```

```
xlabel('Time (msec)'), ylabel('Geophone Output (V)')

% Cross correlation (xcorr)
% There are 10,000 points of data covering 10 divisions of 20 msec each, therefore the
conversion is 0.2/10,000 sec/pt = 2E-5 sec/pt
% lags** gives the x-axis correlation values, ie-- the number of points of the sliding correlation
% index** is the index of the lags** vector where the max correlation occurs
% therefore the time delay between signals is 2E-6 * the lag

conv_factor = .00002;

% Radial correlations
[geo_12r_corr,lags1r] = xcorr(geo_1r,geo_2r,'coeff');
[peak1r,index1r] = max(geo_12r_corr);
delay1r = conv_factor*abs(lags1r(index1r));
speed1r = 2/delay1r;

[geo_23r_corr,lags2r] = xcorr(geo_2r,geo_3r,'coeff');
[peak2r,index2r] = max(geo_23r_corr);
delay2r = conv_factor*abs(lags2r(index2r));
speed2r = 2/delay2r;

[geo_13r_corr,lags3r] = xcorr(geo_1r,geo_3r,'coeff');
[peak3r,index3r] = max(geo_13r_corr);
delay3r = conv_factor*abs(lags3r(index3r));
speed3r = 4/delay3r;

% Vertical correlations
[geo_12v_corr,lags1v] = xcorr(geo_1v,geo_2v,'coeff');
[peak1v,index1v] = max(geo_12v_corr);
delay1v = conv_factor*abs(lags1v(index1v));
speed1v = 2/delay1v;

[geo_23v_corr,lags2v] = xcorr(geo_2v,geo_3v,'coeff');
[peak2v,index2v] = max(geo_23v_corr);
delay2v = conv_factor*abs(lags2v(index2v));
speed2v = 2/delay2v;

[geo_13v_corr,lags3v] = xcorr(geo_1v,geo_3v,'coeff');
[peak3v,index3v] = max(geo_13v_corr);
delay3v = conv_factor*abs(lags3v(index3v));
speed3v = 4/delay3v;

% Calculation of the average radial and vertical speeds
speed_ravg = (speed1r + speed2r + speed3r)/3;
speed_vavg = (speed1v + speed2v + speed3v)/3;

% Radial correlation plots which show the calculated wave speeds
figure(3)
plot(lags1r,geo_12r_corr), title('Radial Correlation between 1m and 3m, Single Source, N=5'),grid
text(lags1r(10000),peak1r-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage radial speed = %2.2f
m/s',lags1r(index1r),peak1r,delay1r,speed1r,speed_ravg))
figure(4)
plot(lags2r,geo_23r_corr), title('Radial Correlation between 3m and 5m, Single Source, N=5'),grid
```

```
text(lags2r(10000),peak2r-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage radial speed = %2.2f
m/s',lags2r(index2r),peak2r,delay2r,speed2r,speed_ravg))
figure(5)
plot(lags3r,geo_13r_corr), title('Radial Correlation between 1m and 5m, Single Source, N=5'),grid
text(lags3r(10000),peak3r-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage radial speed = %2.2f
m/s',lags3r(index3r),peak3r,delay3r,speed3r,speed_ravg))

% Vertical correlation plots which show the calculated wave speeds
figure(6)
plot(lags1v,geo_12v_corr), title('Vertical Correlation between 1m and 3m, Single Source, 100 Hz,
N=5, 80 Vpk-pk, 20 lb'),grid
text(lags1v(10000),peak1v-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage vertical speed = %2.2f
m/s',lags1v(index1v),peak1v,delay1v,speed1v,speed_vavg))
figure(7)
plot(lags2v,geo_23v_corr), title('Vertical Correlation between 3m and 5m, Single Source, 100 Hz,
N=5, 80 Vpk-pk, 20 lb'),grid
text(lags2v(10000),peak2v-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage vertical speed = %2.2f
m/s',lags2v(index2v),peak2v,delay2v,speed2v,speed_vavg))
figure(8)
plot(lags3v,geo_13v_corr), title('Vertical Correlation between 1m and 5m, Single Source,
N=5'),grid
text(lags3v(10000),peak3v-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage vertical speed = %2.2f
m/s',lags3v(index3v),peak3v,delay3v,speed3v,speed_vavg))

% Hankel Plot
figure(10)
plot3(geo_2r,time,geo_2v), grid, axis ij, axis([-.0045 .0045 60 100 -.0045 .0045])
xlabel('Radial'), ylabel('Time (msec)'), zlabel('Vertical'),title('Hankel Plot for 3m, Single Source,
N=5')

% Complex Power ------------------------------------------------------------------------------------------------
% --------------------------------------------------------------------------------------------------------------
complex_1r = hilbert(geo_1r);
complex_1v = hilbert(geo_1v);
complex_power_1 = conj(complex_1v).*complex_1r;
complex_2r = hilbert(geo_2r);
complex_2v = hilbert(geo_2v);
complex_power_2 = conj(complex_2v).*complex_2r;
complex_3r = hilbert(geo_3r);
complex_3v = hilbert(geo_3v);
complex_power_3 = conj(complex_3v).*complex_3r;

% Imaginary Power Plots

figure(16)
plot(time,imag(complex_power_2)), title('Imag Pwr for 3m, Single Source, N=5'), grid, axis([30
140 min(imag(complex_power_2)) max(imag(complex_power_2))])
xlabel('Time (msec)'), ylabel('Imaginary Part of (complex conjugate(radial) * complex(vertical)')
%

% Imaginary Power correlations
```

```
[power_12_corr,lags1e] = xcorr(imag(complex_power_1),imag(complex_power_2),'coeff');
[peak1e,index1e] = max(power_12_corr);
delay1e = conv_factor*abs(lags1e(index1e));
speed1e = 2/delay1e;

[power_23_corr,lags2e] = xcorr(imag(complex_power_2),imag(complex_power_3),'coeff');
[peak2e,index2e] = max(power_23_corr);
delay2e = conv_factor*abs(lags2e(index2e));
speed2e = 2/delay2e;

[power_13_corr,lags3e] = xcorr(imag(complex_power_1),imag(complex_power_3),'coeff');
[peak3e,index3e] = max(power_13_corr);
delay3e = conv_factor*abs(lags3e(index3e));
speed3e = 4/delay3e;

% Calculation of the average speed
speed_eavg = (speed1e + speed2e + speed3e)/3;

% Imaginary Power correlation plots which show the calculated wave speed
figure(20)
plot(lags3e,power_13_corr), title('Imaginary Power Correlation between 1m and 5m, Single
Source, N=5'),grid
text(lags3e(10000),peak3e-.2,sprintf('Max correlation @ (%2.0f,%0.3g)\ntime delay = %0.3g
sec\nwave speed = %2.2f m/s\naverage wave speed = %2.2f
m/s',lags3e(index3e),peak3e,delay3e,speed3e,speed_eavg))
```

B. TIME DELAY VARIATIONS

```
% This program was written specifically for the data collected on
% August 13, 2003. For other data, the load command and variable names must be changed.
clear all

load august13_timedelay

% using 32 pt smoothing, "detrend" levels signal to 0

% G1R50S---> Geophone number 1, Radial, 50 m/s time delay, toward the SAND
% G1R50W---> same as above except toward the WATER

geo_1r_50s=detrend(smooth(G1R50S(:,2),32));
geo_1r_50w=detrend(smooth(G1R50W(:,2),32));
geo_1r_60s=detrend(smooth(G1R60S(:,2),32));
geo_1r_60w=detrend(smooth(G1R60W(:,2),32));
geo_1r_70s=detrend(smooth(G1R70S(:,2),32));
geo_1r_70w=detrend(smooth(G1R70W(:,2),32));
geo_1r_80s=detrend(smooth(G1R80S(:,2),32));
geo_1r_80w=detrend(smooth(G1R80W(:,2),32));
geo_1r_90s=detrend(smooth(G1R90S(:,2),32));
geo_1r_90w=detrend(smooth(G1R90W(:,2),32));
geo_1r_100s=detrend(smooth(G1R100S(:,2),32));
geo_1r_100w=detrend(smooth(G1R100W(:,2),32));
geo_1r_110s=detrend(smooth(G1R110S(:,2),32));
geo_1r_110w=detrend(smooth(G1R110W(:,2),32));
geo_1r_120s=detrend(smooth(G1R120S(:,2),32));
geo_1r_120w=detrend(smooth(G1R120W(:,2),32));
geo_1r_130s=detrend(smooth(G1R130S(:,2),32));
geo_1r_130w=detrend(smooth(G1R130W(:,2),32));
geo_1r_140s=detrend(smooth(G1R140S(:,2),32));
geo_1r_140w=detrend(smooth(G1R140W(:,2),32));

geo_1v_50s=detrend(smooth(G1V50S(:,2),32));
geo_1v_50w=detrend(smooth(G1V50W(:,2),32));
geo_1v_60s=detrend(smooth(G1V60S(:,2),32));
geo_1v_60w=detrend(smooth(G1V60W(:,2),32));
geo_1v_70s=detrend(smooth(G1V70S(:,2),32));
geo_1v_70w=detrend(smooth(G1V70W(:,2),32));
geo_1v_80s=detrend(smooth(G1V80S(:,2),32));
geo_1v_80w=detrend(smooth(G1V80W(:,2),32));
geo_1v_90s=detrend(smooth(G1V90S(:,2),32));
geo_1v_90w=detrend(smooth(G1V90W(:,2),32));
geo_1v_100s=detrend(smooth(G1V100S(:,2),32));
geo_1v_100w=detrend(smooth(G1V100W(:,2),32));
geo_1v_110s=detrend(smooth(G1V110S(:,2),32));
geo_1v_110w=detrend(smooth(G1V110W(:,2),32));
geo_1v_120s=detrend(smooth(G1V120S(:,2),32));
geo_1v_120w=detrend(smooth(G1V120W(:,2),32));
geo_1v_130s=detrend(smooth(G1V130S(:,2),32));
geo_1v_130w=detrend(smooth(G1V130W(:,2),32));
geo_1v_140s=detrend(smooth(G1V140S(:,2),32));
geo_1v_140w=detrend(smooth(G1V140W(:,2),32));

geo_2r_50s=detrend(smooth(G2R50S(:,2),32));
```

```
geo_2r_50w=detrend(smooth(G2R50W(:,2),32));
geo_2r_60s=detrend(smooth(G2R60S(:,2),32));
geo_2r_60w=detrend(smooth(G2R60W(:,2),32));
geo_2r_70s=detrend(smooth(G2R70S(:,2),32));
geo_2r_70w=detrend(smooth(G2R70W(:,2),32));
geo_2r_80s=detrend(smooth(G2R80S(:,2),32));
geo_2r_80w=detrend(smooth(G2R80W(:,2),32));
geo_2r_90s=detrend(smooth(G2R90S(:,2),32));
geo_2r_90w=detrend(smooth(G2R90W(:,2),32));
geo_2r_100s=detrend(smooth(G2R100S(:,2),32));
geo_2r_100w=detrend(smooth(G2R100W(:,2),32));
geo_2r_110s=detrend(smooth(G2R110S(:,2),32));
geo_2r_110w=detrend(smooth(G2R110W(:,2),32));
geo_2r_120s=detrend(smooth(G2R120S(:,2),32));
geo_2r_120w=detrend(smooth(G2R120W(:,2),32));
geo_2r_130s=detrend(smooth(G2R130S(:,2),32));
geo_2r_130w=detrend(smooth(G2R130W(:,2),32));
geo_2r_140s=detrend(smooth(G2R140S(:,2),32));
geo_2r_140w=detrend(smooth(G2R140W(:,2),32));

geo_2v_50s=detrend(smooth(G2V50S(:,2),32));
geo_2v_50w=detrend(smooth(G2V50W(:,2),32));
geo_2v_60s=detrend(smooth(G2V60S(:,2),32));
geo_2v_60w=detrend(smooth(G2V60W(:,2),32));
geo_2v_70s=detrend(smooth(G2V70S(:,2),32));
geo_2v_70w=detrend(smooth(G2V70W(:,2),32));
geo_2v_80s=detrend(smooth(G2V80S(:,2),32));
geo_2v_80w=detrend(smooth(G2V80W(:,2),32));
geo_2v_90s=detrend(smooth(G2V90S(:,2),32));
geo_2v_90w=detrend(smooth(G2V90W(:,2),32));
geo_2v_100s=detrend(smooth(G2V100S(:,2),32));
geo_2v_100w=detrend(smooth(G2V100W(:,2),32));
geo_2v_110s=detrend(smooth(G2V110S(:,2),32));
geo_2v_110w=detrend(smooth(G2V110W(:,2),32));
geo_2v_120s=detrend(smooth(G2V120S(:,2),32));
geo_2v_120w=detrend(smooth(G2V120W(:,2),32));
geo_2v_130s=detrend(smooth(G2V130S(:,2),32));
geo_2v_130w=detrend(smooth(G2V130W(:,2),32));
geo_2v_140s=detrend(smooth(G2V140S(:,2),32));
geo_2v_140w=detrend(smooth(G2V140W(:,2),32));

timeindex = [1:length(geo_1r_50s)];
time = timeindex*.02;
figure(2)
plot(time, geo_2r_70s,'k:', time, geo_2r_80s,'b-', time, geo_2r_90s,'r-.', time, geo_2r_100s,'c--',
time, geo_2r_120s, 'k-.', time, geo_2r_140s, 'm-'),  title('Radial Output at 3.5m, Haversine '),grid
legend('70 m/s time delay', '80 m/s time delay', '90 m/s time delay', '100 m/s time delay', '120 m/s
time delay', '140 m/s time delay')
xlabel('Time (msec)'), ylabel('Geophone Output (V)'), axis([40 100 -.008 .008])
figure(6)
plot(time, geo_2v_70s,'k:', time, geo_2v_80s,'b-', time, geo_2v_90s,'r-.', time, geo_2v_100s,'c--',
time, geo_2v_120s, 'k-.', time, geo_2v_140s, 'm-'),  title('Vertical Output at 3.5m, Haversine'),grid
legend('70 m/s time delay', '80 m/s time delay', '90 m/s time delay', '100 m/s time delay', '120 m/s
time delay', '140 m/s time delay')
xlabel('Time (msec)'), ylabel('Geophone Output (V)'), axis([40 100 -.014 .012])
%
```

```
% Complex Power ------------------------------------------------------------------------------------------------
% ------------------------------------------------------------------------------------------------------------
complex_13rs = hilbert(geo_1r_80s);
complex_13vs = hilbert(geo_1v_80s);
complex_power_13s = conj(complex_13vs).*complex_13rs;
complex_14rs = hilbert(geo_1r_70s);
complex_14vs = hilbert(geo_1v_70s);
complex_power_14s = conj(complex_14vs).*complex_14rs;

complex_1rs = hilbert(geo_1r_90s);
complex_1vs = hilbert(geo_1v_90s);
complex_power_1s = conj(complex_1vs).*complex_1rs;
complex_2rs = hilbert(geo_1r_100s);
complex_2vs = hilbert(geo_1v_100s);
complex_power_2s = conj(complex_2vs).*complex_2rs;
complex_3rs = hilbert(geo_1r_110s);
complex_3vs = hilbert(geo_1v_110s);
complex_power_3s = conj(complex_3vs).*complex_3rs;
complex_4rs = hilbert(geo_1r_120s);
complex_4vs = hilbert(geo_1v_120s);
complex_power_4s = conj(complex_4vs).*complex_4rs;
complex_5rs = hilbert(geo_1r_130s);
complex_5vs = hilbert(geo_1v_130s);
complex_power_5s = conj(complex_5vs).*complex_5rs;
complex_6rs = hilbert(geo_1r_140s);
complex_6vs = hilbert(geo_1v_140s);
complex_power_6s = conj(complex_6vs).*complex_6rs;

complex_13rw = hilbert(geo_1r_80w);
complex_13vw = hilbert(geo_1v_80w);
complex_power_13w = conj(complex_13vw).*complex_13rw;
complex_14rw = hilbert(geo_1r_70w);
complex_14vw = hilbert(geo_1v_70w);
complex_power_14w = conj(complex_14vw).*complex_14rw;

complex_1rw = hilbert(geo_1r_90w);
complex_1vw = hilbert(geo_1v_90w);
complex_power_1w = conj(complex_1vw).*complex_1rw;
complex_2rw = hilbert(geo_1r_100w);
complex_2vw = hilbert(geo_1v_100w);
complex_power_2w = conj(complex_2vw).*complex_2rw;
complex_3rw = hilbert(geo_1r_110w);
complex_3vw = hilbert(geo_1v_110w);
complex_power_3w = conj(complex_3vw).*complex_3rw;
complex_4rw = hilbert(geo_1r_120w);
complex_4vw = hilbert(geo_1v_120w);
complex_power_4w = conj(complex_4vw).*complex_4rw;
complex_5rw = hilbert(geo_1r_130w);
complex_5vw = hilbert(geo_1v_130w);
complex_power_5w = conj(complex_5vw).*complex_5rw;
complex_6rw = hilbert(geo_1r_140w);
complex_6vw = hilbert(geo_1v_140w);
complex_power_6w = conj(complex_6vw).*complex_6rw;


complex_15rs = hilbert(geo_2r_80s);
```

```
complex_15vs = hilbert(geo_2v_80s);
complex_power_15s = conj(complex_15vs).*complex_15rs;
complex_16rs = hilbert(geo_2r_70s);
complex_16vs = hilbert(geo_2v_70s);
complex_power_16s = conj(complex_16vs).*complex_16rs;

complex_7rs = hilbert(geo_2r_90s);
complex_7vs = hilbert(geo_2v_90s);
complex_power_7s = conj(complex_7vs).*complex_7rs;
complex_8rs = hilbert(geo_2r_100s);
complex_8vs = hilbert(geo_2v_100s);
complex_power_8s = conj(complex_8vs).*complex_8rs;
complex_9rs = hilbert(geo_2r_110s);
complex_9vs = hilbert(geo_2v_110s);
complex_power_9s = conj(complex_9vs).*complex_9rs;
complex_10rs = hilbert(geo_2r_120s);
complex_10vs = hilbert(geo_2v_120s);
complex_power_10s = conj(complex_10vs).*complex_10rs;
complex_11rs = hilbert(geo_2r_130s);
complex_11vs = hilbert(geo_2v_130s);
complex_power_11s = conj(complex_11vs).*complex_11rs;
complex_12rs = hilbert(geo_2r_140s);
complex_12vs = hilbert(geo_2v_140s);
complex_power_12s = conj(complex_12vs).*complex_12rs;


complex_15rw = hilbert(geo_2r_80w);
complex_15vw = hilbert(geo_2v_80w);
complex_power_15w = conj(complex_15vw).*complex_15rw;
complex_16rw = hilbert(geo_2r_70w);
complex_16vw = hilbert(geo_2v_70w);
complex_power_16w = conj(complex_16vw).*complex_16rw;

complex_7rw = hilbert(geo_2r_90w);
complex_7vw = hilbert(geo_2v_90w);
complex_power_7w = conj(complex_7vw).*complex_7rw;
complex_8rw = hilbert(geo_2r_100w);
complex_8vw = hilbert(geo_2v_100w);
complex_power_8w = conj(complex_8vw).*complex_8rw;
complex_9rw = hilbert(geo_2r_110w);
complex_9vw = hilbert(geo_2v_110w);
complex_power_9w = conj(complex_9vw).*complex_9rw;
complex_10rw = hilbert(geo_2r_120w);
complex_10vw = hilbert(geo_2v_120w);
complex_power_10w = conj(complex_10vw).*complex_10rw;
complex_11rw = hilbert(geo_2r_130w);
complex_11vw = hilbert(geo_2v_130w);
complex_power_11w = conj(complex_11vw).*complex_11rw;
complex_12rw = hilbert(geo_2r_140w);
complex_12vw = hilbert(geo_2v_140w);
complex_power_12w = conj(complex_12vw).*complex_12rw;

% Imaginary Power Plots
%
figure(23)
```

```
plot(time, imag(complex_power_16s),'k:', time, imag(complex_power_15s),'b-', time,
imag(complex_power_7s),'r-.', time, imag(complex_power_8s),'c--', time,
imag(complex_power_10s), 'k-.', time, imag(complex_power_12s), 'm-'), title('Imag Pwr at 3.5m,
Haversine'),grid
legend('70 m/s time delay', '80 m/s time delay', '90 m/s time delay', '100 m/s time delay', '120 m/s
time delay', '140 m/s time delay')
xlabel('Time (msec)'), ylabel('Imaginary Part of (complex conjugate(vertical) * complex(radial))'),
axis([40 100 -0.00002 .00013])
```

C. ARRAY BEAM PATTERN

```
% This program was written specifically for the data collected on
% August 14, 2003. For other data, the load command and variable names must be changed.
% This program uses the beam pattern data recorded on a radius of 3.5 meters
% at 15 degree increments to calculate relative signal strength, both radial
% and vertical, and relative imaginary pwr as a function of angle.

clear all

load august14_beam_sand    % Array directed toward the sand
load august14_beam_water   % Array directed toward the water
load Calibration_values % values to reference geophones to geo-1 both radially and vertically

% using 32 pt smoothing, most of the noise is removed from the signal
% "detrend" levels signal to 0

% Radial signals
geo_1B0rs=detrend(smooth(G1R000S(:,2),32))/geo_1r_value;
geo_2B15rs=detrend(smooth(G2R015S(:,2),32))/geo_2r_value;
geo_3B30rs=detrend(smooth(G3R030S(:,2),32))/geo_3r_value;
geo_4B45rs=detrend(smooth(G4R045S(:,2),32))/geo_4r_value;
geo_5B60rs=detrend(smooth(G5R060S(:,2),32))/geo_5r_value;
geo_6B75rs=detrend(smooth(G6R075S(:,2),32))/geo_6r_value;
geo_1B90rs=detrend(smooth(G1R090S(:,2),32))/geo_1r_value;
geo_2B165rs=detrend(smooth(G2R165S(:,2),32))/geo_2r_value;
geo_3B150rs=detrend(smooth(G3R150S(:,2),32))/geo_3r_value;
geo_4B135rs=detrend(smooth(G4R135S(:,2),32))/geo_4r_value;
geo_5B120rs=detrend(smooth(G5R120S(:,2),32))/geo_5r_value;
geo_6B105rs=detrend(smooth(G6R105S(:,2),32))/geo_6r_value;
geo_1B180rs=detrend(smooth(G1R180S(:,2),32))/geo_1r_value;
geo_2B195rs=detrend(smooth(G2R195S(:,2),32))/geo_2r_value;
geo_3B210rs=detrend(smooth(G3R210S(:,2),32))/geo_3r_value;
geo_4B225rs=detrend(smooth(G4R225S(:,2),32))/geo_4r_value;
geo_5B240rs=detrend(smooth(G5R240S(:,2),32))/geo_5r_value;
geo_6B255rs=detrend(smooth(G6R255S(:,2),32))/geo_6r_value;
geo_1B270rs=detrend(smooth(G1R270S(:,2),32))/geo_1r_value;
geo_2B345rs=detrend(smooth(G2R345S(:,2),32))/geo_2r_value;
geo_3B330rs=detrend(smooth(G3R330S(:,2),32))/geo_3r_value;
geo_4B315rs=detrend(smooth(G4R315S(:,2),32))/geo_4r_value;
geo_5B300rs=detrend(smooth(G5R300S(:,2),32))/geo_5r_value;
geo_6B285rs=detrend(smooth(G6R285S(:,2),32))/geo_6r_value;

geo_1B0rw=detrend(smooth(G1R000W(:,2),32))/geo_1r_value;
geo_2B15rw=detrend(smooth(G2R015W(:,2),32))/geo_2r_value;
geo_3B30rw=detrend(smooth(G3R030W(:,2),32))/geo_3r_value;
geo_4B45rw=detrend(smooth(G4R045W(:,2),32))/geo_4r_value;
geo_5B60rw=detrend(smooth(G5R060W(:,2),32))/geo_5r_value;
geo_6B75rw=detrend(smooth(G6R075W(:,2),32))/geo_6r_value;
geo_1B90rw=detrend(smooth(G1R090W(:,2),32))/geo_1r_value;
geo_2B165rw=detrend(smooth(G2R165W(:,2),32))/geo_2r_value;
geo_3B150rw=detrend(smooth(G3R150W(:,2),32))/geo_3r_value;
geo_4B135rw=detrend(smooth(G4R135W(:,2),32))/geo_4r_value;
geo_5B120rw=detrend(smooth(G5R120W(:,2),32))/geo_5r_value;
geo_6B105rw=detrend(smooth(G6R105W(:,2),32))/geo_6r_value;
geo_1B180rw=detrend(smooth(G1R180W(:,2),32))/geo_1r_value;
```

```
geo_2B195rw=detrend(smooth(G2R195W(:,2),32))/geo_2r_value;
geo_3B210rw=detrend(smooth(G3R210W(:,2),32))/geo_3r_value;
geo_4B225rw=detrend(smooth(G4R225W(:,2),32))/geo_4r_value;
geo_5B240rw=detrend(smooth(G5R240W(:,2),32))/geo_5r_value;
geo_6B255rw=detrend(smooth(G6R255W(:,2),32))/geo_6r_value;
geo_1B270rw=detrend(smooth(G1R270W(:,2),32))/geo_1r_value;
geo_2B345rw=detrend(smooth(G2R345W(:,2),32))/geo_2r_value;
geo_3B330rw=detrend(smooth(G3R330W(:,2),32))/geo_3r_value;
geo_4B315rw=detrend(smooth(G4R315W(:,2),32))/geo_4r_value;
geo_5B300rw=detrend(smooth(G5R300W(:,2),32))/geo_5r_value;
geo_6B285rw=detrend(smooth(G6R285W(:,2),32))/geo_6r_value;

% Vertical signals
geo_1B0vs=detrend(smooth(G1V000S(:,2),32))/geo_1v_value;
geo_2B15vs=detrend(smooth(G2V015S(:,2),32))/geo_2v_value ;
geo_3B30vs=detrend(smooth(G3V030S(:,2),32))/geo_3v_value;
geo_4B45vs=detrend(smooth(G4V045S(:,2),32))/geo_4v_value;
geo_5B60vs=detrend(smooth(G5V060S(:,2),32))/geo_5v_value;
geo_6B75vs=detrend(smooth(G6V075S(:,2),32))/geo_6v_value;
geo_1B90vs=detrend(smooth(G1V090S(:,2),32))/geo_1v_value;
geo_2B165vs=detrend(smooth(G2V165S(:,2),32))/geo_2v_value ;
geo_3B150vs=detrend(smooth(G3V150S(:,2),32))/geo_3v_value;
geo_4B135vs=detrend(smooth(G4V135S(:,2),32))/geo_4v_value;
geo_5B120vs=detrend(smooth(G5V120S(:,2),32))/geo_5v_value;
geo_6B105vs=detrend(smooth(G6V105S(:,2),32))/geo_6v_value;
geo_1B180vs=detrend(smooth(G1V180S(:,2),32))/geo_1v_value;
geo_2B195vs=detrend(smooth(G2V195S(:,2),32))/geo_2v_value ;
geo_3B210vs=detrend(smooth(G3V210S(:,2),32))/geo_3v_value;
geo_4B225vs=detrend(smooth(G4V225S(:,2),32))/geo_4v_value;
geo_5B240vs=detrend(smooth(G5V240S(:,2),32))/geo_5v_value;
geo_6B255vs=detrend(smooth(G6V255S(:,2),32))/geo_6v_value;
geo_1B270vs=detrend(smooth(G1V270S(:,2),32))/geo_1v_value;
geo_2B345vs=detrend(smooth(G2V345S(:,2),32))/geo_2v_value ;
geo_3B330vs=detrend(smooth(G3V330S(:,2),32))/geo_3v_value;
geo_4B315vs=detrend(smooth(G4V315S(:,2),32))/geo_4v_value;
geo_5B300vs=detrend(smooth(G5V300S(:,2),32))/geo_5v_value;
geo_6B285vs=detrend(smooth(G6V285S(:,2),32))/geo_6v_value;

geo_1B0vw=detrend(smooth(G1V000W(:,2),32))/geo_1v_value;
geo_2B15vw=detrend(smooth(G2V015W(:,2),32))/geo_2v_value ;
geo_3B30vw=detrend(smooth(G3V030W(:,2),32))/geo_3v_value;
geo_4B45vw=detrend(smooth(G4V045W(:,2),32))/geo_4v_value;
geo_5B60vw=detrend(smooth(G5V060W(:,2),32))/geo_5v_value;
geo_6B75vw=detrend(smooth(G6V075W(:,2),32))/geo_6v_value;
geo_1B90vw=detrend(smooth(G1V090W(:,2),32))/geo_1v_value;
geo_2B165vw=detrend(smooth(G2V165W(:,2),32))/geo_2v_value ;
geo_3B150vw=detrend(smooth(G3V150W(:,2),32))/geo_3v_value;
geo_4B135vw=detrend(smooth(G4V135W(:,2),32))/geo_4v_value;
geo_5B120vw=detrend(smooth(G5V120W(:,2),32))/geo_5v_value;
geo_6B105vw=detrend(smooth(G6V105W(:,2),32))/geo_6v_value;
geo_1B180vw=detrend(smooth(G1V180W(:,2),32))/geo_1v_value;
geo_2B195vw=detrend(smooth(G2V195W(:,2),32))/geo_2v_value ;
geo_3B210vw=detrend(smooth(G3V210W(:,2),32))/geo_3v_value;
geo_4B225vw=detrend(smooth(G4V225W(:,2),32))/geo_4v_value;
geo_5B240vw=detrend(smooth(G5V240W(:,2),32))/geo_5v_value;
geo_6B255vw=detrend(smooth(G6V255W(:,2),32))/geo_6v_value;
```

```
geo_1B270vw=detrend(smooth(G1V270W(:,2),32))/geo_1v_value;
geo_2B345vw=detrend(smooth(G2V345W(:,2),32))/geo_2v_value ;
geo_3B330vw=detrend(smooth(G3V330W(:,2),32))/geo_3v_value;
geo_4B315vw=detrend(smooth(G4V315W(:,2),32))/geo_4v_value;
geo_5B300vw=detrend(smooth(G5V300W(:,2),32))/geo_5v_value;
geo_6B285vw=detrend(smooth(G6V285W(:,2),32))/geo_6v_value;

% Cross correlation (xcorr)

% The following uses the max of the correlation of the signal
% with the signal at each 15 degree increment of theta to determine
% relative signal strength.  Division by the max signal
% corrleated with itself normalizes the values.

% Radial correlations
normalizings = max(xcorr(geo_3B30rs,geo_3B30rs));
geo_1B0rs_value = max(xcorr(geo_3B30rs,geo_1B0rs))/normalizings;
geo_2B15rs_value = max(xcorr(geo_3B30rs,geo_2B15rs))/normalizings;
geo_3B30rs_value = max(xcorr(geo_3B30rs,geo_3B30rs))/normalizings;
geo_4B45rs_value = max(xcorr(geo_3B30rs,geo_4B45rs))/normalizings;
geo_5B60rs_value = max(xcorr(geo_3B30rs,geo_5B60rs))/normalizings;
geo_6B75rs_value = max(xcorr(geo_3B30rs,geo_6B75rs))/normalizings;
geo_1B90rs_value = max(xcorr(geo_3B30rs,geo_1B90rs))/normalizings;
geo_2B165rs_value = max(xcorr(geo_3B30rs,geo_2B165rs))/normalizings;
geo_3B150rs_value = max(xcorr(geo_3B30rs,geo_3B150rs))/normalizings;
geo_4B135rs_value = max(xcorr(geo_3B30rs,geo_4B135rs))/normalizings;
geo_5B120rs_value = max(xcorr(geo_3B30rs,geo_5B120rs))/normalizings;
geo_6B105rs_value = max(xcorr(geo_3B30rs,geo_6B105rs))/normalizings;
geo_1B180rs_value = max(xcorr(geo_3B30rs,geo_1B180rs))/normalizings;
geo_2B195rs_value = max(xcorr(geo_3B30rs,geo_2B195rs))/normalizings;
geo_3B210rs_value = max(xcorr(geo_3B30rs,geo_3B210rs))/normalizings;
geo_4B225rs_value = max(xcorr(geo_3B30rs,geo_4B225rs))/normalizings;
geo_5B240rs_value = max(xcorr(geo_3B30rs,geo_5B240rs))/normalizings;
geo_6B255rs_value = max(xcorr(geo_3B30rs,geo_6B255rs))/normalizings;
geo_1B270rs_value = max(xcorr(geo_3B30rs,geo_1B270rs))/normalizings;
geo_2B345rs_value = max(xcorr(geo_3B30rs,geo_2B345rs))/normalizings;
geo_3B330rs_value = max(xcorr(geo_3B30rs,geo_3B330rs))/normalizings;
geo_4B315rs_value = max(xcorr(geo_3B30rs,geo_4B315rs))/normalizings;
geo_5B300rs_value = max(xcorr(geo_3B30rs,geo_5B300rs))/normalizings;
geo_6B285rs_value = max(xcorr(geo_3B30rs,geo_6B285rs))/normalizings;

normalizingw = max(xcorr(geo_3B150rw,geo_3B150rw));
geo_1B0rw_value = max(xcorr(geo_3B150rw,geo_1B0rw))/normalizingw;
geo_2B15rw_value = max(xcorr(geo_3B150rw,geo_2B15rw))/normalizingw;
geo_3B30rw_value = max(xcorr(geo_3B150rw,geo_3B30rw))/normalizingw;
geo_4B45rw_value = max(xcorr(geo_3B150rw,geo_4B45rw))/normalizingw;
geo_5B60rw_value = max(xcorr(geo_3B150rw,geo_5B60rw))/normalizingw;
geo_6B75rw_value = max(xcorr(geo_3B150rw,geo_6B75rw))/normalizingw;
geo_1B90rw_value = max(xcorr(geo_3B150rw,geo_1B90rw))/normalizingw;
geo_2B165rw_value = max(xcorr(geo_3B150rw,geo_2B165rw))/normalizingw;
geo_3B150rw_value = max(xcorr(geo_3B150rw,geo_3B150rw))/normalizingw;
geo_4B135rw_value = max(xcorr(geo_3B150rw,geo_4B135rw))/normalizingw;
geo_5B120rw_value = max(xcorr(geo_3B150rw,geo_5B120rw))/normalizingw;
geo_6B105rw_value = max(xcorr(geo_3B150rw,geo_6B105rw))/normalizingw;
geo_1B180rw_value = max(xcorr(geo_3B150rw,geo_1B180rw))/normalizingw;
geo_2B195rw_value = max(xcorr(geo_3B150rw,geo_2B195rw))/normalizingw;
```

```
geo_3B210rw_value = max(xcorr(geo_3B150rw,geo_3B210rw))/normalizingw;
geo_4B225rw_value = max(xcorr(geo_3B150rw,geo_4B225rw))/normalizingw;
geo_5B240rw_value = max(xcorr(geo_3B150rw,geo_5B240rw))/normalizingw;
geo_6B255rw_value = max(xcorr(geo_3B150rw,geo_6B255rw))/normalizingw;
geo_1B270rw_value = max(xcorr(geo_3B150rw,geo_1B270rw))/normalizingw;
geo_2B345rw_value = max(xcorr(geo_3B150rw,geo_2B345rw))/normalizingw;
geo_3B330rw_value = max(xcorr(geo_3B150rw,geo_3B330rw))/normalizingw;
geo_4B315rw_value = max(xcorr(geo_3B150rw,geo_4B315rw))/normalizingw;
geo_5B300rw_value = max(xcorr(geo_3B150rw,geo_5B300rw))/normalizingw;
geo_6B285rw_value = max(xcorr(geo_3B150rw,geo_6B285rw))/normalizingw;

radial_beams(1) = geo_1B0rs_value;
radial_beams(2) = geo_2B15rs_value;
radial_beams(3) = geo_3B30rs_value;
radial_beams(4) = geo_4B45rs_value;
radial_beams(5) = geo_5B60rs_value;
radial_beams(6) = geo_6B75rs_value;
radial_beams(7) = geo_1B90rs_value;
radial_beams(8) = geo_6B105rs_value;
radial_beams(9) = geo_5B120rs_value;
radial_beams(10) = geo_4B135rs_value;
radial_beams(11) = geo_3B150rs_value;
radial_beams(12) = geo_2B165rs_value;
radial_beams(13) = geo_1B180rs_value;
radial_beams(14) = geo_2B195rs_value;
radial_beams(15) = geo_3B210rs_value;
radial_beams(16) = geo_4B225rs_value;
radial_beams(17) = geo_5B240rs_value;
radial_beams(18) = geo_6B255rs_value;
radial_beams(19) = geo_1B270rs_value;
radial_beams(20) = geo_6B285rs_value;
radial_beams(21) = geo_5B300rs_value;
radial_beams(22) = geo_4B315rs_value;
radial_beams(23) = geo_3B330rs_value;
radial_beams(24) = geo_2B345rs_value;
radial_beams(25) = geo_1B0rs_value;
radial_beam_dBs = 20*log10(radial_beams);
radial_beam_polars = max(radial_beam_dBs + 40,0);

radial_beamw(1) = geo_1B0rw_value;
radial_beamw(2) = geo_2B15rw_value;
radial_beamw(3) = geo_3B30rw_value;
radial_beamw(4) = geo_4B45rw_value;
radial_beamw(5) = geo_5B60rw_value;
radial_beamw(6) = geo_6B75rw_value;
radial_beamw(7) = geo_1B90rw_value;
radial_beamw(8) = geo_6B105rw_value;
radial_beamw(9) = geo_5B120rw_value;
radial_beamw(10) = geo_4B135rw_value;
radial_beamw(11) = geo_3B150rw_value;
radial_beamw(12) = geo_2B165rw_value;
radial_beamw(13) = geo_1B180rw_value;
radial_beamw(14) = geo_2B195rw_value;
radial_beamw(15) = geo_3B210rw_value;
radial_beamw(16) = geo_4B225rw_value;
radial_beamw(17) = geo_5B240rw_value;
```

```
radial_beamw(18) = geo_6B255rw_value;
radial_beamw(19) = geo_1B270rw_value;
radial_beamw(20) = geo_6B285rw_value;
radial_beamw(21) = geo_5B300rw_value;
radial_beamw(22) = geo_4B315rw_value;
radial_beamw(23) = geo_3B330rw_value;
radial_beamw(24) = geo_2B345rw_value;
radial_beamw(25) = geo_1B0rw_value;
radial_beam_dBw = 20*log10(radial_beamw);
radial_beam_polarw = max(radial_beam_dBw + 40,0);

% Vertical correlations
normalizingvs = max(xcorr(geo_1B0vs,geo_1B0vs));
geo_1B0vs_value = max(xcorr(geo_1B0vs,geo_1B0vs))/normalizingvs;
geo_2B15vs_value = max(xcorr(geo_1B0vs,geo_2B15vs))/normalizingvs;
geo_3B30vs_value = max(xcorr(geo_1B0vs,geo_3B30vs))/normalizingvs;
geo_4B45vs_value = max(xcorr(geo_1B0vs,geo_4B45vs))/normalizingvs;
geo_5B60vs_value = max(xcorr(geo_1B0vs,geo_5B60vs))/normalizingvs;
geo_6B75vs_value = max(xcorr(geo_1B0vs,geo_6B75vs))/normalizingvs;
geo_1B90vs_value = max(xcorr(geo_1B0vs,geo_1B90vs))/normalizingvs;
geo_2B165vs_value = max(xcorr(geo_1B0vs,geo_2B165vs))/normalizingvs;
geo_3B150vs_value = max(xcorr(geo_1B0vs,geo_3B150vs))/normalizingvs;
geo_4B135vs_value = max(xcorr(geo_1B0vs,geo_4B135vs))/normalizingvs;
geo_5B120vs_value = max(xcorr(geo_1B0vs,geo_5B120vs))/normalizingvs;
geo_6B105vs_value = max(xcorr(geo_1B0vs,geo_6B105vs))/normalizingvs;
geo_1B180vs_value = max(xcorr(geo_1B0vs,geo_1B180vs))/normalizingvs;
geo_2B195vs_value = max(xcorr(geo_1B0vs,geo_2B195vs))/normalizingvs;
geo_3B210vs_value = max(xcorr(geo_1B0vs,geo_3B210vs))/normalizingvs;
geo_4B225vs_value = max(xcorr(geo_1B0vs,geo_4B225vs))/normalizingvs;
geo_5B240vs_value = max(xcorr(geo_1B0vs,geo_5B240vs))/normalizingvs;
geo_6B255vs_value = max(xcorr(geo_1B0vs,geo_6B255vs))/normalizingvs;
geo_1B270vs_value = max(xcorr(geo_1B0vs,geo_1B270vs))/normalizingvs;
geo_2B345vs_value = max(xcorr(geo_1B0vs,geo_2B345vs))/normalizingvs;
geo_3B330vs_value = max(xcorr(geo_1B0vs,geo_3B330vs))/normalizingvs;
geo_4B315vs_value = max(xcorr(geo_1B0vs,geo_4B315vs))/normalizingvs;
geo_5B300vs_value = max(xcorr(geo_1B0vs,geo_5B300vs))/normalizingvs;
geo_6B285vs_value = max(xcorr(geo_1B0vs,geo_6B285vs))/normalizingvs;

normalizingvw = max(xcorr(geo_1B180vw,geo_1B180vw));
geo_1B0vw_value = max(xcorr(geo_1B180vw,geo_1B0vw))/normalizingvw;
geo_2B15vw_value = max(xcorr(geo_1B180vw,geo_2B15vw))/normalizingvw;
geo_3B30vw_value = max(xcorr(geo_1B180vw,geo_3B30vw))/normalizingvw;
geo_4B45vw_value = max(xcorr(geo_1B180vw,geo_4B45vw))/normalizingvw;
geo_5B60vw_value = max(xcorr(geo_1B180vw,geo_5B60vw))/normalizingvw;
geo_6B75vw_value = max(xcorr(geo_1B180vw,geo_6B75vw))/normalizingvw;
geo_1B90vw_value = max(xcorr(geo_1B180vw,geo_1B90vw))/normalizingvw;
geo_2B165vw_value = max(xcorr(geo_1B180vw,geo_2B165vw))/normalizingvw;
geo_3B150vw_value = max(xcorr(geo_1B180vw,geo_3B150vw))/normalizingvw;
geo_4B135vw_value = max(xcorr(geo_1B180vw,geo_4B135vw))/normalizingvw;
geo_5B120vw_value = max(xcorr(geo_1B180vw,geo_5B120vw))/normalizingvw;
geo_6B105vw_value = max(xcorr(geo_1B180vw,geo_6B105vw))/normalizingvw;
geo_1B180vw_value = max(xcorr(geo_1B180vw,geo_1B180vw))/normalizingvw;
geo_2B195vw_value = max(xcorr(geo_1B180vw,geo_2B195vw))/normalizingvw;
geo_3B210vw_value = max(xcorr(geo_1B180vw,geo_3B210vw))/normalizingvw;
geo_4B225vw_value = max(xcorr(geo_1B180vw,geo_4B225vw))/normalizingvw;
geo_5B240vw_value = max(xcorr(geo_1B180vw,geo_5B240vw))/normalizingvw;
```

```
geo_6B255vw_value = max(xcorr(geo_1B180vw,geo_6B255vw))/normalizingvw;
geo_1B270vw_value = max(xcorr(geo_1B180vw,geo_1B270vw))/normalizingvw;
geo_2B345vw_value = max(xcorr(geo_1B180vw,geo_2B345vw))/normalizingvw;
geo_3B330vw_value = max(xcorr(geo_1B180vw,geo_3B330vw))/normalizingvw;
geo_4B315vw_value = max(xcorr(geo_1B180vw,geo_4B315vw))/normalizingvw;
geo_5B300vw_value = max(xcorr(geo_1B180vw,geo_5B300vw))/normalizingvw;
geo_6B285vw_value = max(xcorr(geo_1B180vw,geo_6B285vw))/normalizingvw;

vertical_beams(1) = geo_1B0vs_value;
vertical_beams(2) = geo_2B15vs_value;
vertical_beams(3) = geo_3B30vs_value;
vertical_beams(4) = geo_4B45vs_value;
vertical_beams(5) = geo_5B60vs_value;
vertical_beams(6) = geo_6B75vs_value;
vertical_beams(7) = geo_1B90vs_value;
vertical_beams(8) = geo_6B105vs_value;
vertical_beams(9) = geo_5B120vs_value;
vertical_beams(10) = geo_4B135vs_value;
vertical_beams(11) = geo_3B150vs_value;
vertical_beams(12) = geo_2B165vs_value;
vertical_beams(13) = geo_1B180vs_value;
vertical_beams(14) = geo_2B195vs_value;
vertical_beams(15) = geo_3B210vs_value;
vertical_beams(16) = geo_4B225vs_value;
vertical_beams(17) = geo_5B240vs_value;
vertical_beams(18) = geo_6B255vs_value;
vertical_beams(19) = geo_1B270vs_value;
vertical_beams(20) = geo_6B285vs_value;
vertical_beams(21) = geo_5B300vs_value;
vertical_beams(22) = geo_4B315vs_value;
vertical_beams(23) = geo_3B330vs_value;
vertical_beams(24) = geo_2B345vs_value;
vertical_beams(25) = geo_1B0vs_value;
vertical_beam_dBs = 20*log10(vertical_beams);
vertical_beam_polars = max(vertical_beam_dBs + 40,0);

vertical_beamw(1) = geo_1B0vw_value;
vertical_beamw(2) = geo_2B15vw_value;
vertical_beamw(3) = geo_3B30vw_value;
vertical_beamw(4) = geo_4B45vw_value;
vertical_beamw(5) = geo_5B60vw_value;
vertical_beamw(6) = geo_6B75vw_value;
vertical_beamw(7) = geo_1B90vw_value;
vertical_beamw(8) = geo_6B105vw_value;
vertical_beamw(9) = geo_5B120vw_value;
vertical_beamw(10) = geo_4B135vw_value;
vertical_beamw(11) = geo_3B150vw_value;
vertical_beamw(12) = geo_2B165vw_value;
vertical_beamw(13) = geo_1B180vw_value;
vertical_beamw(14) = geo_2B195vw_value;
vertical_beamw(15) = geo_3B210vw_value;
vertical_beamw(16) = geo_4B225vw_value;
vertical_beamw(17) = geo_5B240vw_value;
vertical_beamw(18) = geo_6B255vw_value;
vertical_beamw(19) = geo_1B270vw_value;
vertical_beamw(20) = geo_6B285vw_value;
```

```
vertical_beamw(21) = geo_5B300vw_value;
vertical_beamw(22) = geo_4B315vw_value;
vertical_beamw(23) = geo_3B330vw_value;
vertical_beamw(24) = geo_2B345vw_value;
vertical_beamw(25) = geo_1B0vw_value;
vertical_beam_dBw = 20*log10(vertical_beamw);
vertical_beam_polarw = max(vertical_beam_dBw + 40,0);

theta = [0 15 30 45 60 75 90 105 120 135 150 165 180 195 210 225 240 255 270 285 300 315
330 345 360];
theta_radian = (pi/180) * theta;
theta_polar = theta_radian;

% Theoretical Beam Patterns
%
N = 4;
f = 100;
w = 2*pi*f;
c = 100;
d = .25;
k = w/c;
x1 = -.375;
x2 = -.125;
x3 = .125;
x4 = .375;
phi_1 = 0;
phi_2 = d*k;
phi_3 = 2*d*k;
phi_4 = 3*d*k;

angle = [0:1:360];
angle_radian = angle*pi/180;

n = 1;
for n = 1:length(angle)
   Beam(n)          =          (1/N)*(exp(i*(2*pi*cos(angle_radian(n)-pi)*x1+phi_1))          +
exp(i*(2*pi*cos(angle_radian(n)-pi)*x2+phi_2)) + exp(i*(2*pi*cos(angle_radian(n)-pi)*x3+phi_3)) +
exp(i*(2*pi*cos(angle_radian(n)-pi)*x4+phi_4)));
end

BeamdB = max(20*log10(abs(Beam)/max(abs(Beam)))+40,0);

m = 1;
for m = 1:length(angle)
   Beamw(m)          =          (1/N)*(exp(i*(2*pi*cos(angle_radian(m))*x1+phi_1))          +
exp(i*(2*pi*cos(angle_radian(m))*x2+phi_2))   +   exp(i*(2*pi*cos(angle_radian(m))*x3+phi_3))   +
exp(i*(2*pi*cos(angle_radian(m))*x4+phi_4)));
end

BeamdBw = max(20*log10(abs(Beamw)/max(abs(Beamw)))+40,0);

% Radial Beam Pattern plots
figure(3)
polar(theta_polar,radial_beam_polars, 'r'), axis tight
hold on
polar(angle_radian, BeamdB, 'b'), axis tight
```

```
hold off
% title('Radial Signal Beam Pattern in dB at 4m for 4-Source Array, 90 m/s time delay, 100 Hz,
N=5, 80 Vpk-pk, 20 lb')
figure(4)
polar(theta_polar,radial_beam_polarw, 'r'), axis tight
hold on
polar(angle_radian, BeamdBw, 'b'), axis tight
hold off
% title('Radial Signal Beam Pattern in dB at 4m for 4-Source Array, 90 m/s time delay, 100 Hz,
N=5, 80 Vpk-pk, 20 lb')

% Vertical Beam Pattern plots
figure(7)
polar(theta_polar,vertical_beam_polars, 'r'), axis tight
hold on
polar(angle_radian, BeamdB, 'b'), axis tight
hold off
% title('Vertical Signal Beam Pattern in dB at 4m for 4-Source Array, 90 m/s time delay, 100 Hz,
N=5, 80 Vpk-pk, 20 lb')
figure(8)
polar(theta_polar,vertical_beam_polarw, 'r'), axis tight
hold on
polar(angle_radian, BeamdBw, 'b'), axis tight
hold off

% Complex Power ----------------------------------------------------------------------------------
% ------------------------------------------------------------------------------------------------
complex_1rs = hilbert(geo_1B0rs);
complex_1vs = hilbert(geo_1B0vs);
complex_power_1s = conj(complex_1rs).*complex_1vs;
complex_2rs = hilbert(geo_2B15rs);
complex_2vs = hilbert(geo_2B15vs);
complex_power_2s = conj(complex_2rs).*complex_2vs;
complex_3rs = hilbert(geo_3B30rs);
complex_3vs = hilbert(geo_3B30vs);
complex_power_3s = conj(complex_3rs).*complex_3vs;
complex_4rs = hilbert(geo_4B45rs);
complex_4vs = hilbert(geo_4B45vs);
complex_power_4s = conj(complex_4rs).*complex_4vs;
complex_5rs = hilbert(geo_5B60rs);
complex_5vs = hilbert(geo_5B60vs);
complex_power_5s = conj(complex_5rs).*complex_5vs;
complex_6rs = hilbert(geo_6B75rs);
complex_6vs = hilbert(geo_6B75vs);
complex_power_6s = conj(complex_6rs).*complex_6vs;
complex_7rs = hilbert(geo_1B90rs);
complex_7vs = hilbert(geo_1B90vs);
complex_power_7s = conj(complex_7rs).*complex_7vs;
complex_8rs = hilbert(geo_6B105rs);
complex_8vs = hilbert(geo_6B105vs);
complex_power_8s = conj(complex_8rs).*complex_8vs;
complex_9rs = hilbert(geo_5B120rs);
complex_9vs = hilbert(geo_5B120vs);
complex_power_9s = conj(complex_9rs).*complex_9vs;
complex_10rs = hilbert(geo_4B135rs);
complex_10vs = hilbert(geo_4B135vs);
```

```
complex_power_10s = conj(complex_10rs).*complex_10vs;
complex_11rs = hilbert(geo_3B150rs);
complex_11vs = hilbert(geo_3B150vs);
complex_power_11s = conj(complex_11rs).*complex_11vs;
complex_12rs = hilbert(geo_2B165rs);
complex_12vs = hilbert(geo_2B165vs);
complex_power_12s = conj(complex_12rs).*complex_12vs;
complex_13rs = hilbert(geo_1B180rs);
complex_13vs = hilbert(geo_1B180vs);
complex_power_13s = conj(complex_13rs).*complex_13vs;
complex_14rs = hilbert(geo_2B195rs);
complex_14vs = hilbert(geo_2B195vs);
complex_power_14s = conj(complex_14rs).*complex_14vs;
complex_15rs = hilbert(geo_3B210rs);
complex_15vs = hilbert(geo_3B210vs);
complex_power_15s = conj(complex_15rs).*complex_15vs;
complex_16rs = hilbert(geo_4B225rs);
complex_16vs = hilbert(geo_4B225vs);
complex_power_16s = conj(complex_16rs).*complex_16vs;
complex_17rs = hilbert(geo_5B240rs);
complex_17vs = hilbert(geo_5B240vs);
complex_power_17s = conj(complex_17rs).*complex_17vs;
complex_18rs = hilbert(geo_6B255rs);
complex_18vs = hilbert(geo_6B255vs);
complex_power_18s = conj(complex_18rs).*complex_18vs;
complex_19rs = hilbert(geo_1B270rs);
complex_19vs = hilbert(geo_1B270vs);
complex_power_19s = conj(complex_19rs).*complex_19vs;
complex_20rs = hilbert(geo_6B285rs);
complex_20vs = hilbert(geo_6B285vs);
complex_power_20s = conj(complex_20rs).*complex_20vs;
complex_21rs = hilbert(geo_5B300rs);
complex_21vs = hilbert(geo_5B300vs);
complex_power_21s = conj(complex_21rs).*complex_21vs;
complex_22rs = hilbert(geo_4B315rs);
complex_22vs = hilbert(geo_4B315vs);
complex_power_22s = conj(complex_22rs).*complex_22vs;
complex_23rs = hilbert(geo_3B330rs);
complex_23vs = hilbert(geo_3B330vs);
complex_power_23s = conj(complex_23rs).*complex_23vs;
complex_24rs = hilbert(geo_2B345rs);
complex_24vs = hilbert(geo_2B345vs);
complex_power_24s = conj(complex_24rs).*complex_24vs;

complex_1rw = hilbert(geo_1B0rw);
complex_1vw = hilbert(geo_1B0vw);
complex_power_1w = conj(complex_1rw).*complex_1vw;
complex_2rw = hilbert(geo_2B15rw);
complex_2vw = hilbert(geo_2B15vw);
complex_power_2w = conj(complex_2rw).*complex_2vw;
complex_3rw = hilbert(geo_3B30rw);
complex_3vw = hilbert(geo_3B30vw);
complex_power_3w = conj(complex_3rw).*complex_3vw;
complex_4rw = hilbert(geo_4B45rw);
complex_4vw = hilbert(geo_4B45vw);
complex_power_4w = conj(complex_4rw).*complex_4vw;
```

```
complex_5rw = hilbert(geo_5B60rw);
complex_5vw = hilbert(geo_5B60vw);
complex_power_5w = conj(complex_5rw).*complex_5vw;
complex_6rw = hilbert(geo_6B75rw);
complex_6vw = hilbert(geo_6B75vw);
complex_power_6w = conj(complex_6rw).*complex_6vw;
complex_7rw = hilbert(geo_1B90rw);
complex_7vw = hilbert(geo_1B90vw);
complex_power_7w = conj(complex_7rw).*complex_7vw;
complex_8rw = hilbert(geo_6B105rw);
complex_8vw = hilbert(geo_6B105vw);
complex_power_8w = conj(complex_8rw).*complex_8vw;
complex_9rw = hilbert(geo_5B120rw);
complex_9vw = hilbert(geo_5B120vw);
complex_power_9w = conj(complex_9rw).*complex_9vw;
complex_10rw = hilbert(geo_4B135rw);
complex_10vw = hilbert(geo_4B135vw);
complex_power_10w = conj(complex_10rw).*complex_10vw;
complex_11rw = hilbert(geo_3B150rw);
complex_11vw = hilbert(geo_3B150vw);
complex_power_11w = conj(complex_11rw).*complex_11vw;
complex_12rw = hilbert(geo_2B165rw);
complex_12vw = hilbert(geo_2B165vw);
complex_power_12w = conj(complex_12rw).*complex_12vw;
complex_13rw = hilbert(geo_1B180rw);
complex_13vw = hilbert(geo_1B180vw);
complex_power_13w = conj(complex_13rw).*complex_13vw;
complex_14rw = hilbert(geo_2B195rw);
complex_14vw = hilbert(geo_2B195vw);
complex_power_14w = conj(complex_14rw).*complex_14vw;
complex_15rw = hilbert(geo_3B210rw);
complex_15vw = hilbert(geo_3B210vw);
complex_power_15w = conj(complex_15rw).*complex_15vw;
complex_16rw = hilbert(geo_4B225rw);
complex_16vw = hilbert(geo_4B225vw);
complex_power_16w = conj(complex_16rw).*complex_16vw;
complex_17rw = hilbert(geo_5B240rw);
complex_17vw = hilbert(geo_5B240vw);
complex_power_17w = conj(complex_17rw).*complex_17vw;
complex_18rw = hilbert(geo_6B255rw);
complex_18vw = hilbert(geo_6B255vw);
complex_power_18w = conj(complex_18rw).*complex_18vw;
complex_19rw = hilbert(geo_1B270rw);
complex_19vw = hilbert(geo_1B270vw);
complex_power_19w = conj(complex_19rw).*complex_19vw;
complex_20rw = hilbert(geo_6B285rw);
complex_20vw = hilbert(geo_6B285vw);
complex_power_20w = conj(complex_20rw).*complex_20vw;
complex_21rw = hilbert(geo_5B300rw);
complex_21vw = hilbert(geo_5B300vw);
complex_power_21w = conj(complex_21rw).*complex_21vw;
complex_22rw = hilbert(geo_4B315rw);
complex_22vw = hilbert(geo_4B315vw);
complex_power_22w = conj(complex_22rw).*complex_22vw;
complex_23rw = hilbert(geo_3B330rw);
complex_23vw = hilbert(geo_3B330vw);
```

```
complex_power_23w = conj(complex_23rw).*complex_23vw;
complex_24rw = hilbert(geo_2B345rw);
complex_24vw = hilbert(geo_2B345vw);
complex_power_24w = conj(complex_24rw).*complex_24vw;

% Cross correlation (xcorr)

% The following uses the max of the correlation of the imaginary pwr
% with the signal at each 15 degree increment of theta to determine relative signal
% strength.
% Division by the max of the signal corrleated with itself normalizes the values.

% Imaginary Power correlations
normalizingpwrs = max(xcorr(imag(complex_power_3s),imag(complex_power_3s)));
power_0_values                                                              =
max(xcorr(imag(complex_power_3s),imag(complex_power_1s)))/normalizingpwrs;
power_15_values                                                             =
max(xcorr(imag(complex_power_3s),imag(complex_power_2s)))/normalizingpwrs;
power_30_values                                                             =
max(xcorr(imag(complex_power_3s),imag(complex_power_3s)))/normalizingpwrs;
power_45_values                                                             =
max(xcorr(imag(complex_power_3s),imag(complex_power_4s)))/normalizingpwrs;
power_60_values                                                             =
max(xcorr(imag(complex_power_3s),imag(complex_power_5s)))/normalizingpwrs;
power_75_values                                                             =
max(xcorr(imag(complex_power_3s),imag(complex_power_6s)))/normalizingpwrs;
power_90_values                                                             =
max(xcorr(imag(complex_power_3s),imag(complex_power_7s)))/normalizingpwrs;
power_105_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_8s)))/normalizingpwrs;
power_120_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_9s)))/normalizingpwrs;
power_135_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_10s)))/normalizingpwrs;
power_150_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_11s)))/normalizingpwrs;
power_165_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_12s)))/normalizingpwrs;
power_180_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_13s)))/normalizingpwrs;
power_195_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_14s)))/normalizingpwrs;
power_210_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_15s)))/normalizingpwrs;
power_225_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_16s)))/normalizingpwrs;
power_240_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_17s)))/normalizingpwrs;
power_255_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_18s)))/normalizingpwrs;
power_270_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_19s)))/normalizingpwrs;
power_285_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_20s)))/normalizingpwrs;
power_300_values                                                            =
max(xcorr(imag(complex_power_3s),imag(complex_power_21s)))/normalizingpwrs;
```

```
power_315_values =
max(xcorr(imag(complex_power_3s),imag(complex_power_22s)))/normalizingpwrs;
power_330_values =
max(xcorr(imag(complex_power_3s),imag(complex_power_23s)))/normalizingpwrs;
power_345_values =
max(xcorr(imag(complex_power_3s),imag(complex_power_24s)))/normalizingpwrs;
power_360_values = power_0_values;

imag_pwr_beams(1) = power_0_values;
imag_pwr_beams(2) = power_15_values;
imag_pwr_beams(3) = power_30_values;
imag_pwr_beams(4) = power_45_values;
imag_pwr_beams(5) = power_60_values;
imag_pwr_beams(6) = power_75_values;
imag_pwr_beams(7) = power_90_values;
imag_pwr_beams(8) = power_105_values;
imag_pwr_beams(9) = power_120_values;
imag_pwr_beams(10) = power_135_values;
imag_pwr_beams(11) = power_150_values;
imag_pwr_beams(12) = power_165_values;
imag_pwr_beams(13) = power_180_values;
imag_pwr_beams(14) = power_195_values;
imag_pwr_beams(15) = power_210_values;
imag_pwr_beams(16) = power_225_values;
imag_pwr_beams(17) = power_240_values;
imag_pwr_beams(18) = power_255_values;
imag_pwr_beams(19) = power_270_values;
imag_pwr_beams(20) = power_285_values;
imag_pwr_beams(21) = power_300_values;
imag_pwr_beams(22) = power_315_values;
imag_pwr_beams(23) = power_330_values;
imag_pwr_beams(24) = power_345_values;
imag_pwr_beams(25) = power_360_values;
imag_pwr_beam_dBs = 10*log10(imag_pwr_beams);
imag_pwr_beam_polars = max(imag_pwr_beam_dBs + 40,0);

normalizingpwrw = max(xcorr(imag(complex_power_11w),imag(complex_power_11w)));
power_0_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_1w)))/normalizingpwrw;
power_15_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_2w)))/normalizingpwrw;
power_30_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_3w)))/normalizingpwrw;
power_45_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_4w)))/normalizingpwrw;
power_60_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_5w)))/normalizingpwrw;
power_75_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_6w)))/normalizingpwrw;
power_90_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_7w)))/normalizingpwrw;
power_105_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_8w)))/normalizingpwrw;
power_120_valuew =
max(xcorr(imag(complex_power_11w),imag(complex_power_9w)))/normalizingpwrw;
```

```
power_135_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_10w)))/normalizingpwrw;
power_150_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_11w)))/normalizingpwrw;
power_165_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_12w)))/normalizingpwrw;
power_180_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_13w)))/normalizingpwrw;
power_195_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_14w)))/normalizingpwrw;
power_210_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_15w)))/normalizingpwrw;
power_225_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_16w)))/normalizingpwrw;
power_240_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_17w)))/normalizingpwrw;
power_255_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_18w)))/normalizingpwrw;
power_270_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_19w)))/normalizingpwrw;
power_285_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_20w)))/normalizingpwrw;
power_300_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_21w)))/normalizingpwrw;
power_315_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_22w)))/normalizingpwrw;
power_330_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_23w)))/normalizingpwrw;
power_345_valuew                                                                      =
max(xcorr(imag(complex_power_11w),imag(complex_power_24w)))/normalizingpwrw;
power_360_valuew = power_0_valuew;

imag_pwr_beamw(1) = power_0_valuew;
imag_pwr_beamw(2) = power_15_valuew;
imag_pwr_beamw(3) = power_30_valuew;
imag_pwr_beamw(4) = power_45_valuew;
imag_pwr_beamw(5) = power_60_valuew;
imag_pwr_beamw(6) = power_75_valuew;
imag_pwr_beamw(7) = power_90_valuew;
imag_pwr_beamw(8) = power_105_valuew;
imag_pwr_beamw(9) = power_120_valuew;
imag_pwr_beamw(10) = power_135_valuew;
imag_pwr_beamw(11) = power_150_valuew;
imag_pwr_beamw(12) = power_165_valuew;
imag_pwr_beamw(13) = power_180_valuew;
imag_pwr_beamw(14) = power_195_valuew;
imag_pwr_beamw(15) = power_210_valuew;
imag_pwr_beamw(16) = power_225_valuew;
imag_pwr_beamw(17) = power_240_valuew;
imag_pwr_beamw(18) = power_255_valuew;
imag_pwr_beamw(19) = power_270_valuew;
imag_pwr_beamw(20) = power_285_valuew;
imag_pwr_beamw(21) = power_300_valuew;
imag_pwr_beamw(22) = power_315_valuew;
imag_pwr_beamw(23) = power_330_valuew;
imag_pwr_beamw(24) = power_345_valuew;
```

```
imag_pwr_beamw(25) = power_360_valuew;
imag_pwr_beam_dBw = 10*log10(imag_pwr_beamw);
imag_pwr_beam_polarw = max(imag_pwr_beam_dBw + 40,0);

% Imaginary Power Beam Pattern plots
figure(11)
polar(theta_polar,imag_pwr_beam_polars, 'r'), axis tight
hold on
polar(angle_radian, BeamdB, 'b'), axis tight
hold off
figure(12)
polar(theta_polar,imag_pwr_beam_polarw, 'r'), axis tight
hold on
polar(angle_radian, BeamdBw, 'b'), axis tight
hold off

timeindex = [1:length(geo_1B0rs)];
time = timeindex*.02;

% Hankel PLOT
figure(13)
plot3(geo_1B0rs,time,geo_1B0vs), grid, axis ij, axis([-.015 .015 50 80 -.015 .015])
xlabel('Radial'), ylabel('Time (msec)'), zlabel('Vertical'),title('Hankel Plot for 3.5m, Array,
Haversine')
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Ari Ben-Menahem and Sarva Jit Singh, *Seismic Waves and Sources*, Springer-Verlag New York Inc., 1981.

2. Brown G.C. & Musset A.E., *The Inaccessible Earth*, George Allen and Unwin Ltd., 1981.

3. Dobrin, M.B., and Savit, C.H., *Introduction to Geophysical Prospecting,* 4th ed., McGraw-Hill Book Company, 1988.

4. Dolphin, Lambert, "Ground Penetrating Radar Usage and Limitations", [http://www.ldolphin.org/GPRLimits.html], November 1997, Accessed 01 September 2003.

5. Fitzpatrick, Sean M., *Source Development for a Seismo-Acoustic SONAR*, Master's Thesis, Naval Postgraduate School, Monterey, California, December 1998.

6. Gaghan, Frederick E., *Discrete Mode Source Development and Testing For New Seismo-Acoustic SONAR*, Master's Thesis, Naval Postgraduate School, Monterey, California, March 1998.

7. Hall, Patrick W., *Detection and Target-Strength Measurements of Buried Objects Using a Seismo-Acoustic Sonar*, Master's thesis, Naval Postgraduate School, Monterey, California, December 1998.

8. MacLean, Douglas J., *Mobile Source Development for Seismic-SONAR Based Landmine Detection*, Master's Thesis, Naval Postgraduate School, Monterey, California, June 2003.

9. McClelland, Scott C., *A Rolling Line Source for a Seismic SONAR*, Master's Thesis, Naval Postgraduate School, Monterey, California, June 2002.

10. Seismology Lab, California Institute of Technology, [http://www.gps.caltech .edu /~polet/ body_waves.html], Accessed 01 September 2003.

11. Sheetz, Kraig E., *Advancements in Buried Mine Detection Using Seismic SONAR*, Master's Thesis, Naval Postgraduate School, Monterey, California, December 2002.

12. Stewart, William F., *Buried Object Detection Using Surface Waves*, Master's Thesis, Naval Postgraduate School, Monterey, California, September 1995.

13. White, J.E., *Underground Sound Application of Seismic Waves,* Elsevier Science Publishers B.V., 1983.

14. Zamora, George, “Detecting Land Mines.” [http://www.nmt.edu/mainpage /news/landmine.html]. September 1997, Accessed 01 September 2003.

15. Ziomek, Lawrence J., *Fundamentals of Acoustic Field Theory and Space-Time Signal Processing*, CRC Press, Inc., 2000.

16. Matlab Version 6.5.0.180913a Release 13 (Signal Processing Toolbox), June 18, 2002, The Mathworks Inc., Natick, MA.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   8725 John J. Kingman Road, Suite 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Professor Steven R. Baker, Code PH/Ba
   Department of Physics
   Naval Postgraduate School
   Monterey, California

4. Professor Thomas G. Muir
   Department of Physics and Astronomy
   The University of Mississippi
   University, Mississippi

5. Don Brutzman
   Department of Undersea Warfare
   Naval Postgraduate School
   Monterey, California

6. LT Steven Rumph
   Naval Postgraduate School
   Monterey, California

7. ENS Douglas J. MacLean
   Naval Postgraduate School
   Monterey, California

8. Dr. Douglas Toderoff
   Office of Naval Research, Code 322W
   800 N. Quincy St.
   Arlington, Virginia