

DRAFT 1.0

User's Guide to Geoacoustic Inversion using a Rotated Coordinate System

Laurie T. Fialkowski, Joseph F. Lingeitch, and John S. Perkins

June 23, 2002

1 Introduction

This report documents the theory and numerical implementation of a method to invert underwater acoustic measurements for geo-acoustic parameters. The inversion is accomplished by searching for the parameter values that produce the best match between simulated data and observed data. An error function, usually called the cost function, is defined to measure the difference between the observed data and the simulated data. Since the number of unknown parameters is often large, searching for the optimal values which minimize the cost function can be time consuming. Additionally, the parameter landscape (the surface of the cost function, as a function of the parameters) is usually riddled with local minima that must be avoided. The simulated annealing search method presented here is designed to overcome the problem of local minima, which are often mistaken for global minima when gradient optimization methods are employed.

Another difficulty often encountered for a limited data set is that the optimal solution may not be unique. Some parameters may be coupled (their values can be changed in a coordinated way without affecting the cost function), or the cost function may not be sensitive to the values of some parameters. To address this difficulty, an option to search for the optimal parameter values using a rotated coordinate system is included. In addition to speeding the search, this rotated coordinate system provides an indication of the pa-

rameter couplings and the parameter hierarchy (*i.e.* which parameters can be reliably resolved and which cannot). In obtaining a solution, it is important to know which parameters have been uniquely determined, and which are only known in an “effective manner;” the parameter couplings and hierarchy are good indicators of this information.

The only way to overcome problems with non-uniqueness is to use more data. However, for some applications, an “effective” set of parameters might be sufficient. Typically, a data set obtained with a fixed source/receiver geometry at a single frequency with a high signal-to-noise ratio is used in the inversion. If necessary, more parameters can be determined by processing additional frequencies. This User’s Guide describes a method for combining several single-frequency inversions to obtain estimates for a larger set of parameters than is usually possible with a single frequency. An alternative approach would be to define a wide-band cost function. In either case, the coordinate rotation provides an indication as to which parameters can be uniquely determined.

In the next Section an outline of the theory is presented; it relies heavily on previous publications. Section 3 describes the computer implementation and provides an accompanying illustrative example. Sample input files are included in Appendix A, “Makefiles” and installation instructions are in Appendix B, and descriptions of the FORTRAN codes are in Appendix C. Copies of the FORTRAN codes are available from an anonymous ftp site [www.acoustics.nrl.navy.mil].

2 Theory

The simulated annealing optimization technique implemented is discussed in detail in [Collins *et al.*, J. Acoust. Soc. Am. **92**, p. 2770]. The procedure is an iterative one, started by initializing values for all of the unknown parameters, using a forward propagation model to compute a simulated data set, and evaluating the cost function. In each subsequent iteration, the parameters are perturbed, the forward propagation model again computes a simulated data set based on the new values, the cost function is evaluated and the perturbed parameter values are accepted or rejected in a probabilistic manner. If the cost function is reduced, the perturbed values are always accepted. If the cost function is increased, the new values might still be accepted following evaluation of a random function that depends on the size of the increase, and an

artificial “temperature” parameter related to the iteration number. If the increase in the cost function value is large, or the iteration number high, the new parameter values are less likely to be accepted. This possibility of accepting “uphill” moves through the parameter landscape helps to prevent the search from getting stuck in local minima. Because the problem of local minima is so common, it is wise to repeat the multi-iteration simulated annealing search between 10 and 20 times; each search should have a different set of initial parameter values or a different seed for the random number generator, so that the search progresses through a different sampling of the parameter space Ω .

In the Subsections that follow, we discuss the different stages in the iteration process: (1) perturbation of the parameters, (2) forward modeling, (3) the cost function, and (4) accepting/rejecting the perturbation. A fifth Subsection describes the method for combining single-frequency results.

2.1 Perturbation of Parameters

The geo-acoustic environment is described by a set of parameter values, \mathbf{x} . These are conveniently grouped by layers, each including a layer thickness and the sound speed, density, and attenuation at the top of the layer. The sound speed, density, and attenuation values for the bottom of the layer are assumed to be the same as the values for the top of the next layer. This, in effect, produces a continuous profile for sound speed, density, and attenuation. In order to introduce a discontinuity without changing the source code, one should simply include an artificial “thin” layer (\approx one-tenth of a wavelength) between true layers. The bottom layer is a “basement,” or half-space, with constant sound speed, density and attenuation.

For each parameter x_i , there must be a corresponding minimum value, a_i , and maximum value, b_i . This set of minimum and maximum values define the parameter search space $\Omega = \{\mathbf{x} | a_i < x_i < b_i\}$. Generally, these bounds should be made as tight as possible (using *a priori* information). If the interval is defined so tight as to exclude the actual value of a parameter, then the search results will most likely be unreliable, depending on the sensitivity of the cost function to said parameter.

The code provides two options for the perturbation stage of the process: (1) use the “original” coordinate system and (2) use an optimal coordinate system. We recommend the latter option.

An optimally rotated coordinate system provides an efficient means for a

global optimization algorithm to navigate a parameter landscape. Additionally, the technique provides a way to analyze the parameter hierarchy and parameter couplings. As described in the previous Section, this analysis provides the user with an idea of which parameters are reliably resolved.

The cost function to be optimized (minimized) is a non-negative function $E(\mathbf{d}, \mathbf{u}(\mathbf{x}))$, where \mathbf{d} is the measured acoustic data and \mathbf{u} is the simulated data computed based on the geo-acoustic environment described by the parameter values, \mathbf{x} . From here-on, the cost function is simply referred to as E . We will discuss the cost function in more detail in Subsection 2.3.

The optimally rotated coordinate system for the minimization of E is parallel to the eigenvectors of the covariance matrix $\nabla E(\nabla E)^t$ integrated over the parameter space [Collins and Fishman, J. Acoust. Soc. Am. **98**, p. 1637]. The gradient of the cost function, ∇E , is evaluated over the parameter space $\Omega = \{\mathbf{x} | a_i < x_i < b_i\}$, where a_i and b_i are the bounds for the i th geo-acoustic parameter, x_i , and $1 \leq i \leq n$. More succinctly, the covariance matrix of interest is

$$K = \int_{\Omega} \nabla E(\nabla E)^t d\Omega, \quad (1)$$

where dimensions of the search parameters are removed by the normalization

$$\hat{x}_i = \frac{x_i}{b_i - a_i}. \quad (2)$$

Most geo-acoustic inversion applications involve large dimensional parameter spaces, and thus K is most efficiently estimated with a Monte Carlo method.

The eigenvectors of K , or $\{\mathbf{v}_j\}$, identify the average cost function gradient directions for the parameter space, and also define a new orthonormal basis for the parameter space. Therefore, optimization by navigation of the parameter landscape along the rotated coordinate system is essentially a search for the coefficients $\{y_j\}$ of $\{\mathbf{v}_j\}$ which minimize E . The geo-acoustic parameter set \mathbf{x} is related to \mathbf{y} by

$$\mathbf{x} = \sum_j y_j \mathbf{v}_j. \quad (3)$$

With a single frequency inversion, it is rare for all elements of \mathbf{y} to converge, which implies that not all geo-acoustic parameters ($\{x_i\}$) are reliably resolved.

Each parameter is randomly perturbed according to

$$\hat{x}'_i = \hat{x}_i + \frac{1}{2}\gamma_j^3 \mathbf{v}_j, \quad (4)$$

where $\gamma_j \in [-1, 1]$ is a uniformly distributed random number and $\{\mathbf{v}_j\}$ is the set of normalized eigenvectors of K . Cubing the random number provides for small perturbations as most likely, with large perturbations still possible. If a perturbed parameter value is generated that lies outside of the limiting interval, the value is reflected back into the interval [Collins *et al.*, J. Acoust. Soc. Am. **92**, p. 2770]. It should be noted here that when the option to use the original coordinate system is chosen, the vectors $\{\mathbf{v}_j\}$ are replaced with the canonical basis vectors. That is, the i -th component of $\{\mathbf{v}_j\}$ is zero if $i \neq j$ and is one if $i = j$.

For each eigenvector perturbation to $\hat{\mathbf{x}}$, the cost function is evaluated, and either accepted or rejected based on the acceptance criteria of the simulated annealing algorithm being implemented (see Section 2.4). The acceptance criteria is a function of the cooling schedule. The parameter set $\hat{\mathbf{x}}'$ which produces the lowest value for E over a predetermined number of iterations is the estimate of the optimal parameters for the data set in question. For the inversion algorithm described here, E is evaluated for each eigenvector perturbation.

In addition to defining the coordinate rotation, the eigenvectors of K also disclose the correlations between parameters. These parameter correlations can most readily be identified by plotting $\{\mathbf{v}_j\}$. Any non-zero entry in an eigenvector \mathbf{v}_j indicates that the parameter associated with that entry is correlated to other parameters with non-zero entries in the same eigenvector.

A completely uncorrelated parameter space would yield the standard basis vectors, as illustrated in Fig. 1(A). Figure 1(B) illustrates a case where the first two parameters are correlated with each other, while the third parameter remains uncorrelated from the rest. Furthermore, the ordering of the eigenvectors $\{\mathbf{v}_j\}$ by the size of each associated eigenvalue illustrates the parameter hierarchy, with the eigenvector with the largest associated eigenvalue being first and at the bottom of each plot. The eigenvectors of K with the highest associated eigenvalues are often referred to as the “top eigenvectors.” The relative sizes of the eigenvalues of K indicate the relative sizes of the average cost function gradient in the rotated coordinates. The “top eigenvectors” identify the parameter combinations which most significantly affect the cost function value. Similarly, the directions (or eigenvectors) associated with the smallest

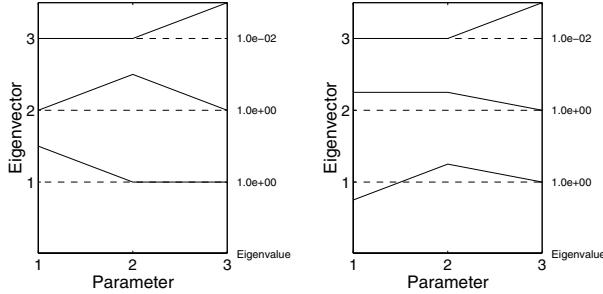


Figure 1: (A) Three completely uncorrelated parameters. (B) Parameters 1 and 2 are correlated; parameter 3 is uncorrelated.

eigenvalues have little effect on the cost function. In Fig. 1, the example eigenvalues of K are noted in a column to the right of the eigenvector plots. In Fig. 1(A), all parameters are uncorrelated, and parameters 1 and 2 are more significant to the value of E than parameter 3. In Fig. 1(B), parameters 1 and 2 are correlated and more significant to the value of E than parameter 3. Parameters which only contribute to the lowest eigenvectors may be held fixed at reasonable values, or taken out of the inversion problem in some other way, as their values have a small impact on the value of E . This decreases the necessary computation time for the inversion and increases the reliability of the results. It is also important to note that only the parameter combinations which are contained in the top eigenvectors can be reliably resolved via inversion. A concrete example of this will be presented in Section 3.

It is important to note that the covariance matrix K depends on both the cost function E and the search space Ω . If a change is made that effects E (such as changing frequency or source position) or Ω (such as adding parameters or changing parameter windows) then K must be recomputed prior to performing the inversion. It is not necessary, however, to recompute K if a new search is initiated using a new initial guess for the parameter set or a new seed for the random number generator.

Another important consequence of the dependence of K on Ω is that the ranking of a particular parameter in the parameter hierarchy (and therefore its resolvability) depends not only on the bounding interval for that parameter, but also on the bounding intervals for *all* of the parameters.

Inversions using different data sets (for example different frequencies) can

be combined using a least-squares method which takes parameter hierarchy into account. This method will be described in Section 2.5.

2.2 Forward Modeling

For the forward modeling we use the RAMGEO code available via anonymous ftp at www.acoustics.nrl.navy.mil. This is a slight variation of the original Range-dependent Acoustic Model (RAM) based upon the split-step Padé solution for the parabolic equation [Collins, J. Acoust. Soc. Am. **93**, p. 1736]. The variation is that the sediment layers are defined relative to the bathymetry, instead of the ocean surface. This is a more useful option for geoacoustic inversion, where the sediment structure is often defined by layer thicknesses.

2.3 Cost Function

The cost function supplied with the code matches normalized complex pressure for the case of a vertical array, and matches amplitudes for the case of a horizontal array:

$$E(\mathbf{d}, \mathbf{u}(\mathbf{x})) = \sqrt{1.0 - |\mathbf{u}(\mathbf{x}) \cdot \mathbf{d}|^2}. \quad (5)$$

Here $\mathbf{u}(\mathbf{x})$ is the modeled normalized complex pressure (vertical array) or the modeled amplitude (horizontal array), and \mathbf{d} is the measured data of the same form. Other appropriate non-negative cost functions may be used in place of eq. 5; only a simple modification of the subroutine *energy.f* is necessary for incorporation into this package.

2.4 Accepting or Rejecting Perturbations

For the code presented here, each perturbation to \mathbf{x} is accepted if

$$\beta \exp\left(\frac{\Delta E}{T}\right) \leq 1.0, \quad (6)$$

where β is a random number uniformly distributed between 0 and 1, $\Delta E = E(\hat{\mathbf{x}}') - E(\hat{\mathbf{x}})$, and T is the simulated temperature, which is lowered with each iteration according to a chosen cooling schedule. The cooling schedule chosen for *inversion.f* is $T = 1/iter$, where *iter* is the iteration counter. This acceptance criteria (eq. 6) always accepts perturbations that decrease the

value of E , and sometimes accepts perturbations which increase the value of E . Allowing the value of E to increase is necessary in order to escape from local minima in the cost function space, and is a main defining feature of a simulated annealing algorithm.

2.5 Consolidation of multi-frequency inversion results

Parameter hierarchies often differ from frequency to frequency. Consequently, when possible, it is beneficial to determine the optimum parameter rotation and perform the geo-acoustic inversion for several very different frequencies. This results in a frequency dependent set of eigenvectors and the corresponding estimates of the optimum coefficients (\mathbf{y}), leading to frequency-dependent estimates of the original geo-acoustic parameters (\mathbf{x}). Ideally, each frequency will have a different parameter hierarchy, and therefore each frequency will reliably resolve a different subset of the geo-acoustic parameters. The multiple sets of results can be combined using a least squares method in order to obtain one estimated value for each geo-acoustic parameter. This consolidated estimate of the geo-acoustic parameter set should yield a relatively low value of E for data at the frequencies used in the inversions.

Let the set $\{\mathbf{w}_i\}$ contain the eigenvectors associated with the largest eigenvalues for all frequencies. If y_i is the estimate of the coefficient of $\{\mathbf{w}_i\}$ found by inversion, then the “consolidated” estimate of the original parameters, \mathbf{x} , is determined from the expression $\mathbf{w}_i^t \mathbf{x} \approx y_i$. In matrix notation, where the rows of A are the vectors $\{\mathbf{w}_i\}$, this is $A\mathbf{x} \approx \mathbf{y}$, and an estimate of \mathbf{x} is found using the singular value decomposition. If $A^t A$ has s small eigenvalues, then the number of resolved parameters is approximately $n - s$. Which parameters are reliably resolved is defined by the parameters that contribute to the “top eigenvectors” of $A^t A$, i.e. those eigenvectors with relatively large associated eigenvalues. An example of this analysis is given in Section 3.4.

3 Implementation with Illustrative Example

This Section contains step-by-step instructions for using the RAMGEO based geo-acoustic inversion with rotated coordinates and for analyzing and understanding the results of each step, accompanied by an illustrative example.

The main program is *inversion*. If a rotated coordinate system is to be

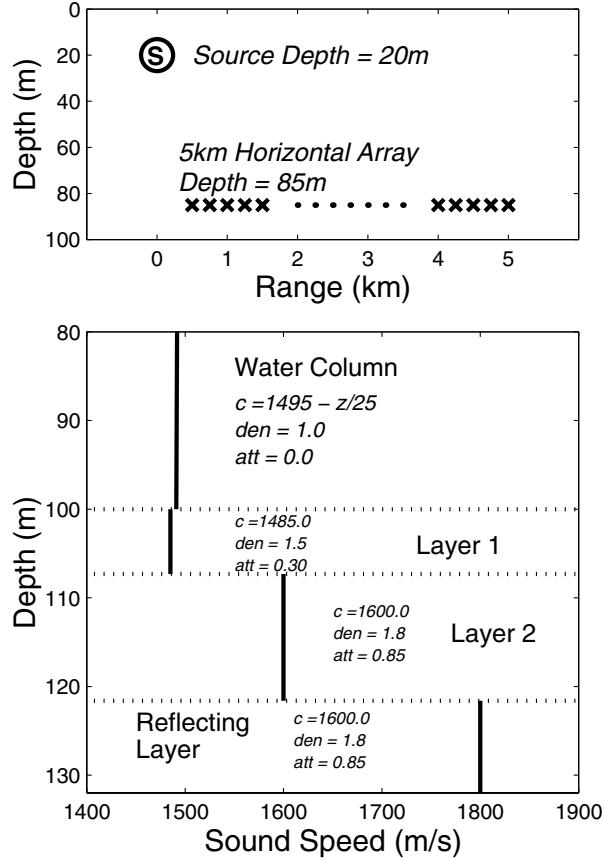


Figure 2: Example problem

used, then *landscape* must be run prior to *inversion*. The input file, *f_grad.in*, should always be the same for the two programs.

The example chosen involves a known source at a depth of 20m and a known horizontal receiver array at a depth of 85m, with phones spanning 5m to 5km in increments of 5m. (This source and receiver arrangement was one of the options used in the *ONR/SPAWAR Inversion Techniques Workshop* [<ftp://itworkshop.nrlssc.navy.mil/pub>].) The example environment involves a flat bottom and two isovelocity sediment layers over a reflecting layer (see Fig. 2).

This example assumes reasonable *a priori* information: a known sound

i	Parameter (x_i)	True	Min (a_i)	Max (b_i)	Δx_i for ∇E
	Layer 1:				
1.	h_1 Thickness (m)	7.30	3.00	30.0	$2dz$
2.	c_1 Sound Speed (m/s)	1485	1450	1600	5.0
3.	ρ_1 Density (gm/cm ³)	1.50	1.0	1.80	0.05
4.	α_1 Attenuation (dB/λ)	0.30	0.10	0.40	0.02
	Layer 2:				
5.	h_2 Thickness (m)	14.3	5.00	50.0	$2dz$
6.	c_2 Sound Speed (m/s)	1600	1550	1700	5.0
7.	ρ_2 Density (gm/cm ³)	1.80	1.30	2.00	0.05
8.	α_2 Attenuation (dB/λ)	0.85	0.10	0.90	0.02
	Reflecting Layer:				
9.	c_3 Sound Speed (m/s)	1800	1700	1900	5.0
10.	ρ_3 Density (gm/cm ³)	2.10	1.80	2.30	0.05
11.	α_3 Attenuation (dB/λ)	0.10	0.01	0.50	0.02

Table 1: Parameterization of the environment for example.

speed profile for the water column, a known bathymetry, and a known number of sediment layers. The remaining unknown parameters define the inversion problem: each sediment layer's thickness, sound speed, density and attenuation. These parameters are detailed in Table 1, along with the parameter bounds. The last column of the Table, “ Δx_i for ∇E ,” refers to the computational step size for each parameter when using a Monte Carlo method to compute ∇E . In this column, notice that $2dz$ is indicated as the computational step size for the thickness parameters; dz is the computational depth grid in RAMGEO, and is a frequency-dependent value. The value of $2dz$ is chosen because a Δx_i smaller than the RAMGEO computational grid size would yield a false zero gradient in that direction.

3.1 Problem definition

The first step in the inversion process is to define the problem by selecting a cost function and the parameters to be inverted for. The cost function is the

same as eqn. 5, and the parameters to be inverted for are those detailed in Table 1.

3.2 Obtaining rotated coordinate system

Once the problem definition has been established, the rotated coordinate system can be estimated. The code that estimates K (eq. 1) is *landscape.f* (named so because it explores the parameter landscape), and its corresponding input file is *f_grad.in*.

The algorithm for *landscape.f* involves sampling the parameter space randomly to compute K by Monte Carlo integration. For each iteration, a point in parameter space Ω is randomly selected, and, each in turn, the following are evaluated for that point: the modeled replica $\mathbf{u}(\mathbf{x})$ (using RAMGEO as a subroutine), the cost function E , and the gradient of the cost function in each direction at that point ∇E . The cumulative sum of $\nabla E(\nabla E)^t$ over all iterations yields the estimate of the covariance matrix K .

As a rule of thumb, 200 iterations is usually sufficient to form a representative covariance matrix K , but for parameter spaces with very largely spaced bounds (*i.e.* $(b_i - a_i)$ is large) more iterations may be necessary to adequately sample the parameter space. The output of *landscape.f* is an ASCII file *sa_eig.out*, which contains eigenvalues and eigenvectors of K . The parameter space Ω has been adequately sampled when the eigenvectors of K do not change substantially as the number of iterations is increased nor do they change substantially when the random number generator seed is changed. Similarly, the eigenvectors should not change substantially for different data sets when source attributes are maintained. For example, if the source is at or close to the same location and at the same frequency, but the data is taken at a different time, then the eigenvectors for K should be similar, assuming the geo-acoustic environment is time-invariant. However, for obtaining optimal results, it is safest to re-compute K for different data sets.

Figure 3 is a plot of the eigenvectors of K for the example problem at 10 Hz, 25 Hz, 100 Hz, and 500 Hz. The associated eigenvalue for each eigenvector is printed in the column to the right of each plot. Note that the eigenvectors are numbered such that the one with the highest associated eigenvalue is first and the one with the lowest associated eigenvalue is last; consequently the eigenvector with the highest associated eigenvalue is at the bottom of the plot.

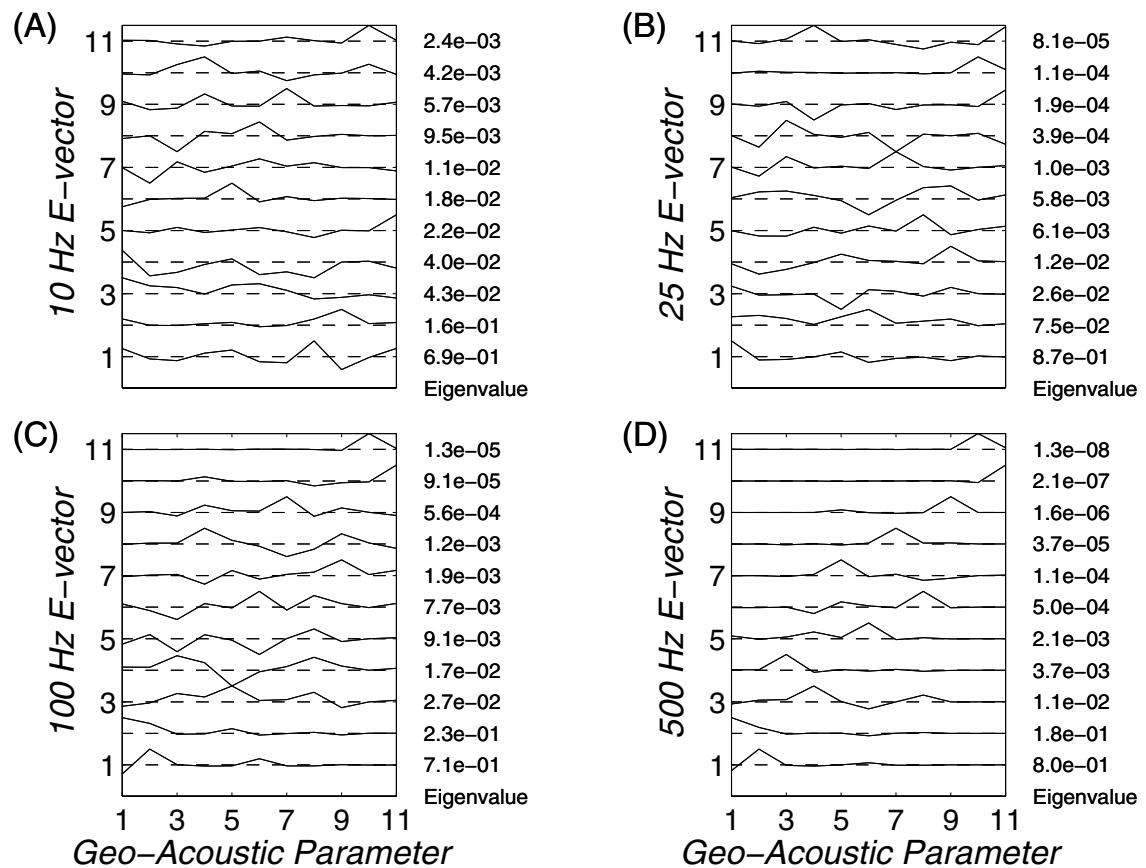


Figure 3: Eigenvectors for K in example problem at (A) 10 Hz, (B) 25 Hz, (C) 100 Hz, and (D) 500 Hz.

The eigenvectors for 500 Hz are very interesting; there is very little correlation between parameters. The most significant eigenvector (“top eigenvector”) for this case has significant contributions from parameters 1 and 2 (the first sediment layer thickness and sound speed). The three least significant eigenvectors only have contributions from parameters 9, 10, and 11 (the sound speed, density and attenuation of the reflecting layer). This parameter hierarchy is not surprising, as higher frequencies do not penetrate as far into the sediments as lower frequencies; it is not expected that higher frequency inversions will resolve deeper sediment layer parameter values. The relevant point of the parameter hierarchy for this example is that inversions at 500 Hz are expected to yield reliable values for parameters 1 and 2, but not necessarily so for parameters 9, 10 and 11. The results of the 500 Hz inversion confirm this expected outcome and can be observed in Table 2. This interpretation of the eigenvectors of K shows how these eigenvectors can be a very valuable tool in any inversion.

3.3 Inversion with the rotated coordinate system

The inversion code is *inversion.f* and the input file is the same *f_grad.in*. The code requires the same data (\mathbf{d}) as is required for estimating K , as well as the eigenvectors for K . For the initialization of $E(\mathbf{x})$, the algorithm randomly selects a point in the parameter space Ω . After evaluating the value of the cost function E for this initial set of parameter values, each subsequent iteration involves the perturbation of the parameters along the rotated coordinates, and the re-evaluation of E at the new point in the parameter space.

Because parameters are generally coupled in a complicated way, it is difficult to discern if a single geo-acoustic parameter is converging when observing the geo-acoustic parameter history. The perturbation history of the rotated coordinates’ coefficients (y_i) are more reliable indicators of convergence. This can be seen by comparing the coefficient perturbation history in Fig. 4 and the parameter perturbation history in Fig. 5; convergence is easier to identify in Fig. 4 than in Fig. 5.

In general, the coefficients for the top eigenvectors of K should converge, while the convergence of the coefficients for those with small associated eigenvalues is less likely, although still possible. In other words, the parameters contained in the top eigenvectors are most likely to be reliably resolved. However, if the same parameters contained in the top eigenvectors also contribute

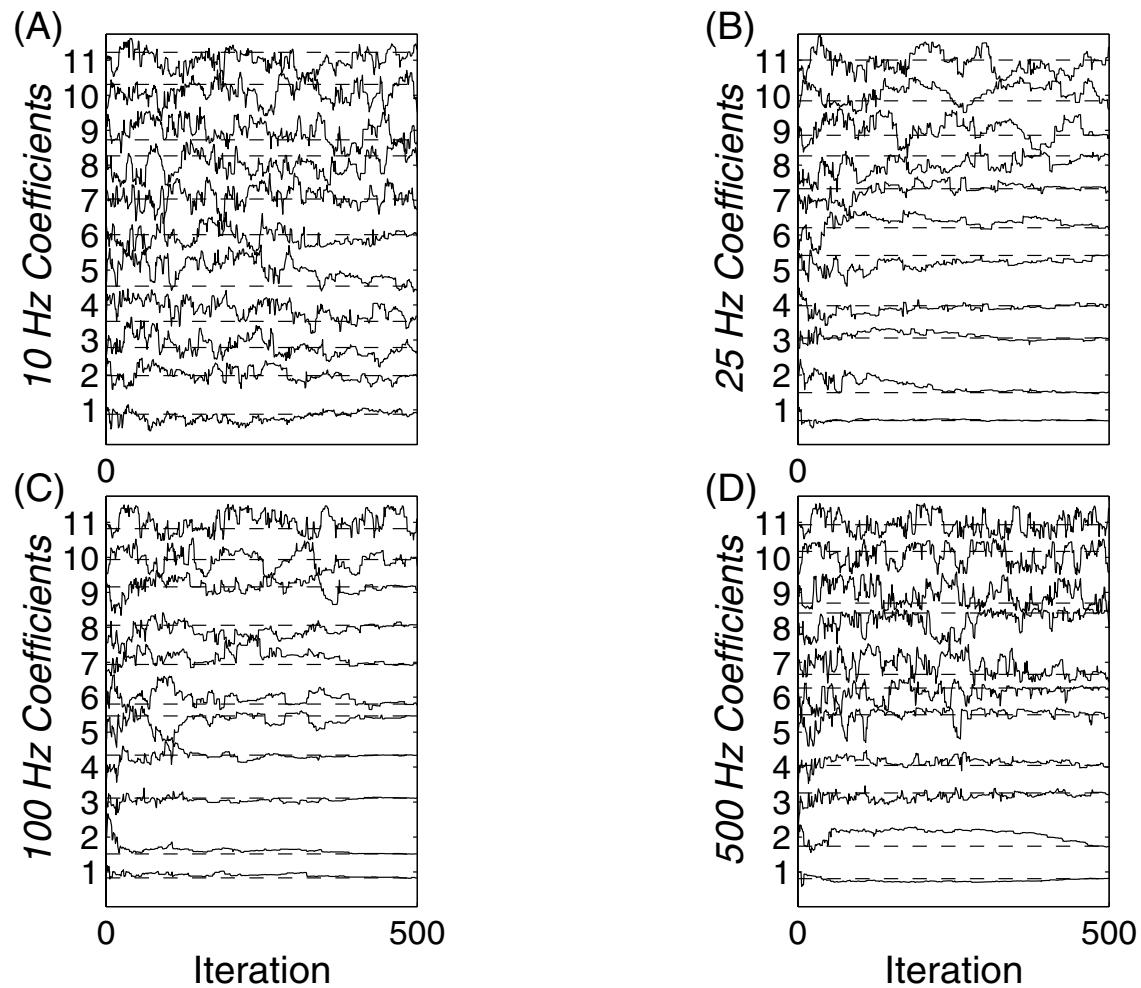


Figure 4: Coefficient histories for inversion using example problem at (A) 10 Hz, (B) 25 Hz, (C) 100 Hz, and (D) 500 Hz. Coefficients are $\{y_j\}$, where $\mathbf{x} = \sum_j y_j \mathbf{v}_j$ with \mathbf{x} the geo-acoustic parameter set and $\{\mathbf{v}_j\}$ the set of eigenvectors of K .

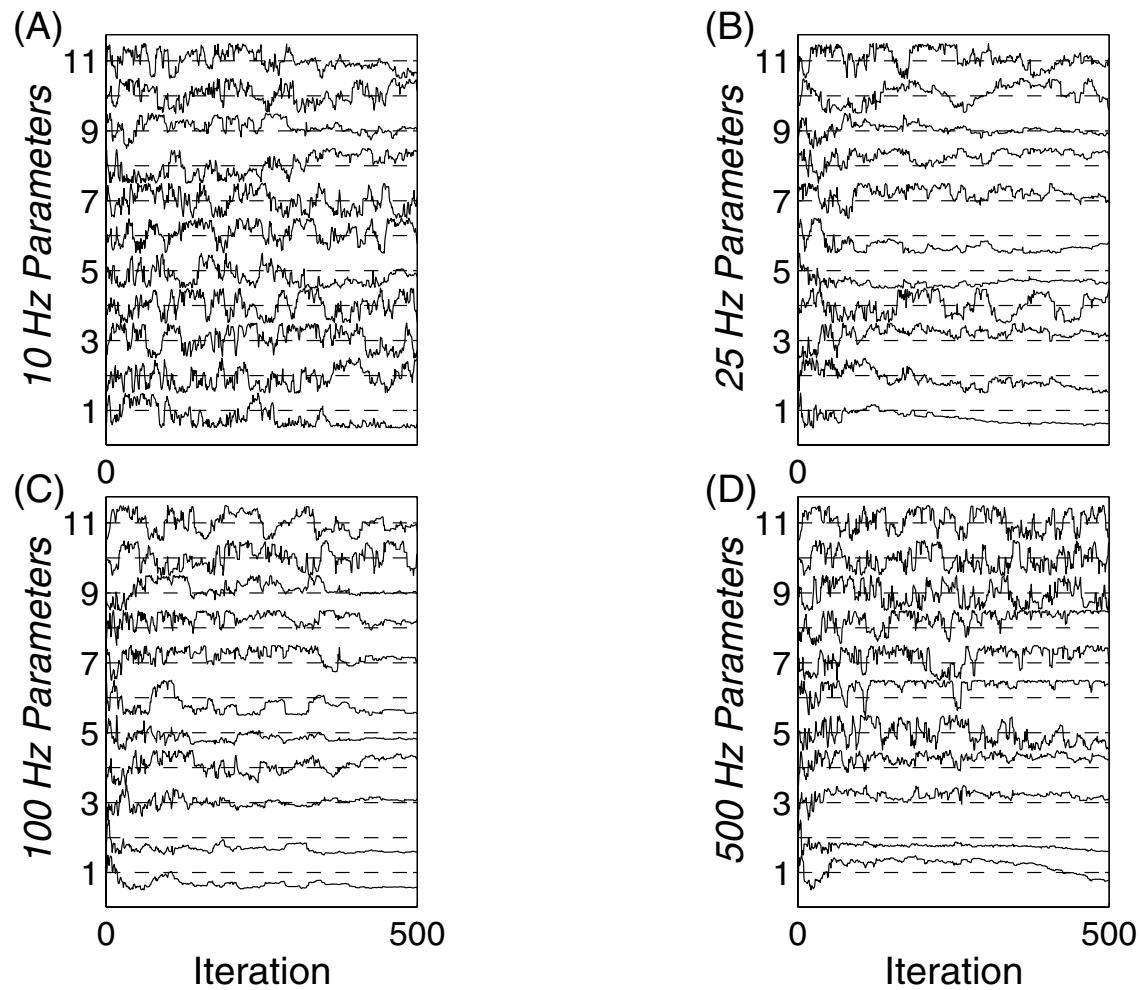


Figure 5: Geo-acoustic parameter histories for inversion using example problem at (A) 10 Hz, (B) 25 Hz, (C) 100 Hz, and (D) 500 Hz.

significantly to eigenvectors with small associated eigenvalues, then these parameters may not be reliably resolved for the set of data in question. To illustrate this caveat, observe the eigenvectors for 25 Hz and for 500 Hz illustrated in Fig. 3. Parameter x_2 (the isovelocity for first sediment layer) appears significantly in eigenvectors 2,4,7 and 8 for the 25 Hz case. If the coefficients for all these eigenvectors were to converge, then x_2 should be reliably resolved. Figure 4 shows that coefficients y_2 and y_4 converge, but it is not clear if y_7 and y_8 converge. Therefore, the value resolved for parameter x_2 is not necessarily reliable for this inversion result. For the 500 Hz case, parameter 2 (x_2) only appears in the first two eigenvectors, and the corresponding coefficients do converge (as seen in Fig. 4); therefore this parameter is more reliably resolved. The inversion results for 25Hz yield an optimal value of 1481.5 m/s for parameter 2, and the inversion results for 500Hz yield a value of 1486.0 m/s for parameter 2. The true value for parameter 2 is 1485.0.

The coefficient histories are output by *inversion.f* to *ffp_inv.out*, while the parameter histories are output to *ffp_inv_prm.out*. The “best parameters” (the parameters that produced the lowest value for E over the course of the inversion) are output to *best.dat*, along with the cost function value and iteration number of occurrence. A word of caution: do not compare actual cost function values over different frequencies as a method to determine if one frequency produces the best inversion results. Higher frequencies tend to have a larger cost function value and this is a side effect of the cost function used.

3.4 Least-Squares multi-frequency consolidation of results

The final step of geo-acoustic inversion using a rotated coordinate system is the consolidation of multiple resultant parameter sets. The example presented here has four separate resultant parameter sets; the results for inversions at 10 Hz, 25 Hz, 100 Hz, and 500 Hz. These four sets of results are detailed in Table 2, along with the consolidated parameter set. Each column of results is accompanied by a column labeled “% err;” where $\% \text{err} = 100 \cdot \frac{\text{val}-\text{true}}{b_i-a_i}$.

The code that performs the singular value decomposition described in Section 2.5 is *least_squares.f*. The code requires the eigenvectors for K for all cases, as well as the “best” parameter sets for each case. The input file is *least_squares.in*. The output file *sa_eig.LS.out* contains the eigenvalues and

Param	True	10 Hz		25 Hz		100 Hz		500 Hz		L.S.	
	Val	Val	%err	Val	%err	Val	%err	Val	%err	Val	%err
1. h_1	7.30	8.28	3.6	6.40	3.3	5.35	7.2	7.36	0.2	5.98	4.9
2. c_1	1485	1535	34	1481	2.4	1488	2.1	1486	1.0	1489	2.5
3. ρ_1	1.50	1.44	7.2	1.66	20	1.46	5.5	1.49	0.7	1.49	1.2
4. α_1	0.30	0.23	25	0.33	9.8	0.33	9.5	0.31	4.1	0.32	6.7
5. h_2	14.3	16.7	5.2	12.7	3.5	17.0	5.9	34.2	44	16.1	4.0
6. c_2	1600	1623	15	1560	27	1574	17	1598	1.0	1583	6.0
7. ρ_2	1.80	1.87	11	1.75	6.8	1.80	0.6	1.94	20	1.82	2.9
8. α_2	0.85	0.82	4.1	0.86	1.7	0.73	15	0.79	7.2	0.72	16
9. c_3	1800	1798	0.8	1802	1.2	1800	0.1	1888	44	1797	1.3
10. ρ_3	2.10	2.25	31	2.18	15	2.25	29	2.00	21	2.05	10
11. α_3	0.10	0.11	2.3	0.11	2.5	0.15	11	0.25	31	0.26	33

Table 2: Results of geo-acoustic inversions at 10 Hz, 25 Hz, 100 Hz, 500 Hz, and the least squares consolidations (L.S.) of all four frequencies. Here the error of parameter measurement is defined in terms of the window for each parameter by $\%err = 100 \cdot \frac{|val - true|}{b_i - a_i}$.

eigenvectors for $A^t A$ and the resulting parameter set is output to the screen.

When running *least_squares*, the user is asked to determine how many eigenvectors should be incorporated into A for each frequency. Here the user should examine the coefficient histories for each frequency (Fig. 4); the number of coefficients that converge is the number of eigenvectors that should be included in A . For the example presented, only y_1 converges for 10 Hz, while $\{y_1 \dots y_4\}$ converge for 25 Hz, $\{y_1 \dots y_9\}$ converge for 100 Hz, and $\{y_1 \dots y_6\}$ converge for 500 Hz. Therefore, in this example, the rows of A are the eigenvectors associated with these coefficients that converged, and therefore A is a 20×11 matrix. After the eigenvalue/eigenvector decomposition of $A^t A$ is computed, the eigenvalues are displayed. The user is then asked to choose how many eigenvectors to use for determining x . The eigenvectors with relatively large eigenvalues should be incorporated, and those with relatively small eigenvalues should be ignored. In the example presented, the eigenvalue/eigenvector decomposition for $A^t A$ yields nine eigenvalues which are greater than one, and two eigenvalues which are close to zero. Therefore, the eigenvectors associated with the top nine eigenvalues are chosen for determining x . The resulting parameter set is shown in the last “Val” column of Table 2. This parameter set defines an “effective environment” for the data used (10Hz, 25Hz, 100Hz, and 400Hz).

A Appendix: Sample input files

A general *f_grad.in* input file is in Fig. 6, followed by a detailed description of each variable in Table 3. Table 3 is a list of the order of the parameters as expected by the code; this parameterization can be changed by modifying the subroutine *parmtr.f*. The input file for the 25Hz example described in Section 3 is in Fig. 7.

f_grad.in				
nter	mter	temp0		
nsamp	iseed	idata		
freq	zs	nrec	dzrec	zrec0
rmin	rmax	dr		
zmax	dz	tabsorb		
c0	np	ns	rs	
nlayer	nbathy			
z_1	cw_1			
z_2	cw_2			
.				
.				
z_n	cw_n			
-1	-1			
ict	$x0_1$	Δx_1	a_1	b_1
.				
.				
.				
ict	$x0_{npar}$	Δx_{npar}	a_{npar}	b_{npar}
-1	-1	-1	-1	-1

Figure 6: Sample input file for example problem. The -1 entries are flags that denote the end of the sound speed profile and the end of the parameterization.

Variable name	Description
<i>f_grad.in</i>	ignored 1 st line; used for comments or title
nter	number of iterations for inversion
mter	number of iterations after which the acceptance criteria are discarded and only improvements to cost function are accepted
temp0	initial temperature; used in cooling schedule
nsamp	number of samples (iterations) for computing K
iseed	integer seed for random number generator <i>rand</i>
idata	type of data 0 ⇒ synthetic vertical array data 1 ⇒ vertical array data read from file <i>pressure.dat</i> 2 ⇒ synthetic horizontal array data 3 ⇒ horizontal array data read from file <i>pressure.dat</i>
freq	frequency (Hz)
zs	depth of source (<i>m</i>)
nrec	number of receiver depths (<i>nrec</i> = 1 if horizontal array)
dzrec	vertical receiver spacing in depth (<i>m</i>)
zrec0	depth of shallowest receiver (depth of all receivers for horizontal array)
rmin	minimum range for
rmax	maximum range for computations and
dr	range step (<i>m</i>) for RAMGEO; varies with frequency Also used as horizontal array receiver spacing; this can be changed by adding separate receiver array variable to code
zmax	“computational universe;” maximum depth (<i>m</i>) for RAMGEO
dz	RAMGEO computational depth grid spacing (<i>m</i>)
tabsorb	thickness of absorption layer at bottom of half-space (in <i>m</i>)
c0	reference soundspeed for RAMGEO
np	number of pade coefficients for RAMGEO
ns	== 1; number of stability constraints; see RAM documentation.
rs	== 0.0; associated with long-range propagation stability constraints; see RAM documentation.

Table 3: Input file variables.

Variable name	Description
nlayer	number of layers for environmental description
nbathy	number of bathymetry points
z_i	i^{th} depth point in water column
cw_i	sound speed (m/s) at z_i
ict	parameter counter; dummy integer variable (not used)
x_0	Default value for x_i , only used when calculating synthetic data, otherwise $\frac{b_i - a_i}{2}$ is used
Δx_i	computational step size for evaluating ∇E
a_i	Minimum bound for parameter i
b_i	Maximum bound for parameter i

Table 4: Continuation of input file variables.

Variable name	Description
$f_{(2 \cdot i - 1)}$	Range (m) for depth point i , $i = 1$ to nbathy
$f_{(2 \cdot i)}$	Depth (m) for depth point i , $i = 1$ to nbathy
$f_{(i+2 \cdot nbathy)}$	Layer thickness (m) for layer i , $i = 1$ to nlayer-1
$f_{(i+1+2 \cdot nbathy)}$	Sound speed (m/s) for layer i , $i = 1$ to nlayer-1
$f_{(i+2+2 \cdot nbathy)}$	Density (gm/cm^3) for layer i , $i = 1$ to nlayer-1
$f_{(i+3+2 \cdot nbathy)}$	Attenuation (dB/λ) for layer i , $i = 1$ to nlayer-1
$f_{(4 \cdot (nlayer - 1) + 1 + 2 \cdot nbathy)}$	Sound speed (m/s) for reflecting layer ($i = nlayer$)
$f_{(4 \cdot (nlayer - 1) + 2 + 2 \cdot nbathy)}$	Density (gm/cm^3) for reflecting layer ($i = nlayer$)
$f_{(4 \cdot (nlayer - 1) + 3 + 2 \cdot nbathy)}$	Attenuation (dB/λ) for reflecting layer ($i = nlayer$)

Table 5: Order of parameters in input file and description.

f_grad.in					
500	500	1.0			nter, mter, temp0
100	1234	2			nsamp iseed idata
25.0	20.0	1	1.0	85.0	freq zs nrec dzrec zrec0
5.0	5000.0	5.0			rmin rmax dr
1250.0	1.0	500.0			zmax dz tabsorb
1500.0	5	1	0.0		c0 np ns rs
3	1				nl nbath
0.0	1495.0				z cw
300.0	1483.0				
-1	-1				
1	0.0	0.0	0.0	0.0	range ₁
2	100.0	0.0	139.0	141.0	depth ₁
3	7.3	1.05	3.0	30.0	thickness ₁
4	1485.0	5.0	1450.0	1600.0	speed ₁
5	1.50	0.05	1.0	1.80	density ₁
6	0.30	0.02	0.1	0.4	att ₁
7	14.3	1.05	5.0	50.0	thickness ₂
8	1600.0	5.0	1550.0	1700.0	speed ₂
9	1.80	0.05	1.3	2.0	density ₂
10	0.85	0.02	0.10	0.9	att ₂
11	1800.0	5.0	1700.0	1900.0	speed ₃
12	2.10	0.05	1.8	2.3	density ₃
13	0.10	0.02	0.01	0.5	att ₃
-1	-1	-1	-1	-1	

Figure 7: Sample input file for example problem at 25Hz.

B Appendix: Make files and installation instructions

C Appendix: FORTRAN code description

There are four main routines, and most have several subroutines in common. The four main routines are *landscape*, *inversion*, *just_energy*, and *least_squares*.

1. *landscape* is the routine that determines the optimal coordinate system for a given data set and parameter set, and *landscape.f* is the calling program. The subroutines for this routine are contained in *parmtr.f*, *ram.f*, *energy.f*, and *sa_eig.f*.
2. *inversion* is the routine that uses the optimal coordinate system computed by *landscape* (or the conventional coordinate system) to navigate the parameter landscape while optimizing a cost function of choice. The subroutines for this routine are contained in *setup.f*, *parmtr.f*, *ram.f*, and *energy.f*.
3. *least_squares* uses a singular value decomposition to determine the parameters with the least squares error using data and coordinate systems for various frequencies (or data sets). The subroutines are *rdinfil*, *geteigs*, *getbest*, *lmats*, *svinv*, *lsqre*, *jacobi*, and *eigsrt*, and are all contained within *least_squares.f*.
4. *just_energy* computes the cost function value for a given set of parameters. This is useful for determining how well parameters perform for different frequencies when determined from inversion at another frequency, or from the consolidation of multiple frequency results.

Following is an alphabetical listing of the subroutines mentioned above, along with a brief description of each.

1. *energy.f* computes the cost function.
2. *parmtr.f* defines the environmental and geometric parameterization.
3. *ram.f* is RAMGEO in a subroutine. All subroutines for RAMGEO have been put into the library *libramgeo.a*.
4. *sa_eig.f* finds the eigenvalues of a covariance matrix. This routine uses *jacobi.f* and *eigsrt.f* from *Numerical Recipes* [Press et al., *Numerical Recipes in FORTRAN* (Cambridge University Press, Cambridge, 1992)].

5. *setup.f* reads an input file and a data file (or computes synthetic data), and sets up arrays necessary for RAMGEO. There is a call to the subroutine *zread*; this routine is contained in the RAMGEO library, *libramgeo.a*.
6. *setup2.f* is very similar to *setup.f*, except it allows the user the opportunity to change some of the values that are read in from *f_grad.in* (so that endless editing of *f_grad.in* is not necessary).