

Global optimization to maximize stack energy

David B. Reister*, J. Barhen*, John B. DuBose[†], and E. M. Oblow*

ABSTRACT

We have developed a new method for global optimization, designed to maximize stack energy. Our method includes upper bounds for the stack energy and maximization of the stack energy for each common midpoint. The method quickly identifies ranges of parameter values where the maximum cannot be located. We have used the approach to reach an excellent solution for a difficult sample problem that was created by adding static disruptions to field data. Our new method should apply in many areas of geophysics.

INTRODUCTION

Since the earliest days of seismic exploration, geophysicists have recognized the need to correct for the low velocity in the weathered and unconsolidated sediments near the earth's surface. The data processing procedure has been described by Yilmaz (1987), Marsden (1993), and Sheriff and Geldart (1995). The conventional method for calculating residual statics corrections was developed in the seminal papers of Taner et al. (1974) and Wiggins et al. (1976). Their method for calculating residual statics corrections has two steps: (1) use crosscorrelation to estimate the total time delay for each trace and (2) use least squares to find the parameters in the total time delay equation. Ronen and Claerbout (1985) proposed a one-step alternative approach: stack energy maximization. They defined an objective function that measures the correlation between all of the traces in each CMP gather. Changes in the parameters in the total time delay equation cause a time shift for each trace and change the correlation between traces. The parameter values are varied to maximize the stack energy.

The stack energy function depends on thousands of traces and hundreds of parameters and can have a very large number of local maxima. Most optimization methods find a local maximum. A problem with many local maxima requires a global op-

timization method. Rothman (1985) recognized that the residual statics problem was a global optimization problem and proposed to solve the problem using the simulated annealing method. Subsequently, Rothman improved his method (1986) and applied it to some field data that required large static corrections (up to 200 ms). DuBose (1993) proposed several innovations to improve Rothman's method.

We have developed an improved algorithm for solving global optimization problems called TRUST (terminal repeller unconstrained subenergy tunneling). Several published papers describe TRUST (Cetin et al., 1993; Barhen and Protopopescu, 1996; Barhen et al., 1997). It is superior to all competing methods for a standard set of nonconvex benchmark problems. However, the stack energy problem is much larger than the benchmark problems, and large problems are more difficult than small ones.

We describe the substantial research required to develop this new global optimization method for the stack energy problem. We define the stack energy, derive an upper bound for the stack energy, and define a stack energy for each CMP. The solution of the CMP problems provides both an initial estimate of the static corrections and a closer upper bound for the total stack energy. A discussion of our new method for global optimization [stochastic Pijavskij tunneling (SPT)] follows. The method quickly identifies ranges of parameter values where the maximum cannot be located. With a resolution of 100 000 possible points in the parameter range, we must usually evaluate the function about ten times to exclude all points. We end the paper with an excellent solution for a difficult test problem based on field data and our conclusions.

THE RESIDUAL STATICS OPTIMIZATION PROBLEM

Stack energy

We consider a common midpoint (CMP) stack of traces that have normal moveout corrections. For each trace ($t = 1, N_t$), we assume that the data consist of the Fourier components ($f = 1, N_f$) of the measured signal. The Fourier components

Manuscript received by the Editor June 1, 1999; revised manuscript received May 23, 2000.

*Oak Ridge National Laboratory, Bldg. 6010, MS 6355, 1 Bethel Valley Road, Oak Ridge, Tennessee 37831-6355. E-mail: dbr@ornl.gov; barhenj@ornl.gov.

†Formerly CogniSeis-Paradigm Geophysical, 1214 Oak Tree, Houston, Texas 77055-6311; presently Geophysical Development Corp., 8401 Westheimer Road, Suite 150, Houston, Texas 77063-2799. E-mail: johnd@geodev.com.

© 2001 Society of Exploration Geophysicists. All rights reserved.

(D_{ft}) are complex numbers. The seismic energy travels from a source (s_t) to a receiver (r_t) via a midpoint (k_t). For each midpoint k , the data are stacked:

$$H_{kf} = \sum_t \exp[2\pi i f (S_{s_t} + R_{r_t})] D_{ft}. \quad (1)$$

The statics corrections (S_s and R_r) are determined to maximize the total energy (E) in the stacked data:

$$E = \sum_k \sum_f |H_{kf}|^2. \quad (2)$$

Coherence factor

Sheriff and Geldart (1995) discuss several measures of coherence in the time domain. We define a coherence factor in the frequency domain. When all of the traces are in phase, the coherence factor equals one. We disaggregate the total energy [equation (2)] by defining the energy for each CMP (E_k):

$$E_k = \sum_f |H_{kf}|^2. \quad (3)$$

Then the total energy is the sum of the E_k . We derive an upper bound (G_k) for each of the E_k . The coherence factor (Q_k) is the ratio of E_k and G_k :

$$Q_k = E_k / G_k. \quad (4)$$

Since the E_k are positive, each of the coherence factors will be in the range (0, 1).

To define the upper bound (G_k), we write the complex numbers D_{ft} in polar form

$$D_{ft} = \alpha_{ft} \exp(i\theta_{ft}) \quad (5)$$

and define the phase (ψ_{ft}) by

$$\psi_{ft} = 2\pi f (S_{s_t} + R_{r_t}) + \theta_{ft}. \quad (6)$$

Using ψ_{ft} , the stacked data [equation (1)] may be written

$$H_{kf} = \sum_t \alpha_{ft} \exp(i\psi_{ft}) \quad (7)$$

and the stack energy for each CMP [equation 3] is given by

$$E_k = \sum_f \sum_t \sum_\tau \alpha_{ft} \alpha_{f\tau} \cos(\psi_{ft} - \psi_{f\tau}). \quad (8)$$

When all of the traces are in phase, $\cos(\psi_{ft} - \psi_{f\tau}) = 1$ for all frequencies and traces in the CMP, and the upper bound on the stack energy is

$$G_k = \sum_f \sum_t \sum_\tau \alpha_{ft} \alpha_{f\tau} = \sum_f \left(\sum_t \alpha_{ft} \right)^2. \quad (9)$$

Maximize stack energy for each CMP

We have defined the energy E_k and a coherence factor Q_k for each CMP. However, each of the statics values can influence many of the CMP gathers. We can decouple the CMP gathers by introducing new variables. The advantage of this approach is that we can solve a number of small problems rather than one large problem and derive a closer upper bound for the stack energy of the coupled problem. We have found that the solutions of the decoupled problems can be mapped to an excellent initial estimate for the parameters for the coupled problem.

For each trace t , we define v_t by

$$v_t = S_{s_t} + R_{r_t}. \quad (10)$$

Using matrix notation, the decoupled parameters satisfy

$$\mathbf{v} = \mathbf{A}\mathbf{x}, \quad (11)$$

where \mathbf{x} is the parameter vector that combines the vector of source statics corrections (\mathbf{S}) and the vector of receiver statics corrections (\mathbf{R}):

$$\mathbf{x} = \begin{bmatrix} \mathbf{S} \\ \mathbf{R} \end{bmatrix}. \quad (12)$$

The \mathbf{A} matrix has many more rows than columns. The number of columns is the total number of source and receiver static corrections, while the number of rows is the total number of traces. All of the elements of each row are zero except for two elements that are one: the source position and the receiver position for the trace. The \mathbf{A} matrix does not have full column rank. Taner et al. (1974) demonstrate that the matrix has a 3-D null space. The three null space vectors are (1) a constant for all sources and zero for receivers, (2) a constant for all receivers and zero for sources, and (3) a vector that is proportional to the source position for the sources and proportional to the receiver position for the receivers. More discussion of the null space vectors is in Reister et al. (1999).

Using the new variables, the stacked data definition [equation (1)] becomes

$$H_{kf} = \sum_t \exp[2\pi i f v_t] D_{ft}. \quad (13)$$

Each H_{kf} depends on a unique subset of v_t (${}^k\mathbf{v}$), and the problem is decoupled. Using the new variables (${}^k\mathbf{v}$), the components of the stack energy are

$$E_k({}^k\mathbf{v}) = \sum_f |H_{kf}|^2. \quad (14)$$

Each CMP stack energy global optimization problem can be solved independently to find the components of the vector \mathbf{v} (v_t). We shall call the maximum value for each CMP stack energy ${}^D G_k$.

The value ${}^D G_k$ is a closer upper bound for the components of the stack energy than G_k . Why is ${}^D G_k$ an upper bound for E_k ? Each source and receiver static correction influences many values of E_k . As each static correction changes, some values of E_k increase and others decrease. Since H_{kf} depends on a unique subset of v_t (${}^k\mathbf{v}$), there are no trade-offs with other E_k , and ${}^D G_k$ is an upper bound for E_k . Why is ${}^D G_k < G_k$? We have demonstrated [equation (8)] that $E_k = G_k$ when all phases ψ_{ft} are equal for all traces in the CMP gather and all frequencies. Since we have only one component of \mathbf{v} to control all of the Fourier components for a trace, we normally are unable to get all of the components in phase.

We have used singular value decomposition to find the inverse for the matrix equation (11) that defines the disaggregated parameters (\mathbf{v}) and map \mathbf{v} back to the static corrections (\mathbf{x}):

$$\mathbf{x} = \mathbf{B}\mathbf{v}. \quad (15)$$

We find that the most striking feature of each row of the \mathbf{B} matrix is that only a few values are large and they are all approximately equal. Furthermore, the large values in a row occur for a trace that has a one in the corresponding column of the \mathbf{A} matrix. We will now demonstrate why the \mathbf{B} matrix has these features and derive an approximate value for the large values of the \mathbf{B} matrix.

Since \mathbf{B} is the inverse matrix: $\mathbf{BA} = \mathbf{I}$. Thus, the dot product of row n of \mathbf{B} with column n of \mathbf{A} is one and the dot product of row n of \mathbf{B} with all other columns of \mathbf{A} is zero. We can approximate \mathbf{B} by assuming that all of the elements in row n are equal to zero except the traces that correspond to the positive values in column n of \mathbf{A} , and that all of the nonzero values in row n of \mathbf{B} are equal to the same value (ρ_n). With this assumption, the dot product of row n of \mathbf{B} with column n of \mathbf{A} is $M_n \rho_n$, where M_n is the number of nonzero values in column n of \mathbf{A} . Since the dot product should be equal to one, the value of ρ_n is $\rho_n = 1/M_n$. We derive the following approximate map from \mathbf{v} to \mathbf{x} :

$$x_n \cong \frac{\sum_t a_{tn} v_t}{M_n}. \quad (16)$$

Thus, we calculate x_n by averaging over all of the components of \mathbf{v} that contain x_n .

Convergence factor

The convergence factor (F_k) is the ratio of the energy for each CMP (E_k) and the maximum CMP energy (${}^D G_k$):

$$F_k = E_k / {}^D G_k. \quad (17)$$

Since E_k is positive, each convergence factor will be in the range (0, 1). Let's compare convergence factor and coherence factor. In the coherence factor, all Fourier components for all traces in a CMP gather are in phase. In the convergence factor, CMP energy reaches its upper bound, and there are values for the static corrections that allow the CMP to reach that bound. The convergence factor allows us to identify which CMPs have the potential to increase their stack energy.

GLOBAL OPTIMIZATION ALGORITHM

To maximize the stack energy, we solve the following optimization problem: Given an objective function [$f(\mathbf{x})$, where \mathbf{x} is an N -dimensional vector constrained to lie in a domain P], find the global maximum (\mathbf{x}_g). That is, find a point \mathbf{x}_g in P such that $f(\mathbf{x}_g) \geq f(\mathbf{x})$ for all \mathbf{x} in P . To introduce the concepts of global optimization, we will consider the 1-D function displayed in Figure 1. The function has five maxima. We can find an approximate value for the global maximum by evaluating the function at M points on a regular grid and choosing the largest value. As we increase M , we will develop a better estimate of the largest value. However, this brute force approach fails for problems with many parameters because the total number of function evaluations increases exponentially (M^N), where N is the number of parameters. We can reduce the number of function evaluations by concentrating them near local maxima.

As discussed in Barhen et al. (1997), Trust finds the global maximum by executing a series of cycles of ascent and shooting (descent and tunneling for a global minimum). Beginning at point x_1 , we ascend to the first local maximum f_1 . We shoot (move in the positive x -direction) until we find a point with

a value larger than f_1 , and then we ascend to the next local maximum. We continue to shoot and ascend until we reach the upper bound on x . The highest value we find is the global maximum.

A rigorous proof (Cetin et al., 1993) shows that Trust will find the global maximum for the 1-D case. If we approximate an N -dimensional problem by a 1-D curve that covers the N -dimensional region P (or by a regular grid), we can formally solve all global optimization problems. However, this method is impractical for large problems because the number of function evaluations increases exponentially with the number of dimensions (the number of evaluations is M^N , where M is the number of function evaluations in each dimension).

Since the number of parameters is very large for the stack energy function, we cannot search a regular grid with M^N points in region P . Fortunately, we have had good success using 1-D paths to search for the global maximum of the stack energy [we have worked on another global optimization problem (protein folding) that requires an exhaustive search of an M^N grid]. From a local maximum, we use a 1-D algorithm to explore each N -dimension of the problem, one at a time (the effort increases linearly with N rather than exponentially). When we find a point where the energy function has a higher value than the previous local maximum, we ascend to the next local maximum. If we explore all N directions and do not find a higher value, we stop the algorithm. The choice of the dimension to search next can be numerical order or random.

We have developed SPT (stochastic Pijavskij tunneling), a new algorithm that greatly speeds up the 1-D shooting phase (tunneling for minimization). The algorithm is stochastic because the function is evaluated at random points rather than at every point on a grid. The algorithm uses Pijavskij cones to eliminate regions of the 1-D space [for a discussion of Pijavskij's algorithm, see Oblow (1999)]. Before we use Figure 1 to discuss SPT, we will discuss two features of the algorithm: ascent to a local maximum and the Lipschitz constant.

For the residual statics optimization problem, calculating the stack energy function [$f(\mathbf{x})$] is expensive (requires many floating point operations) and the calculation of the derivatives of the function with respect to each of the components of the

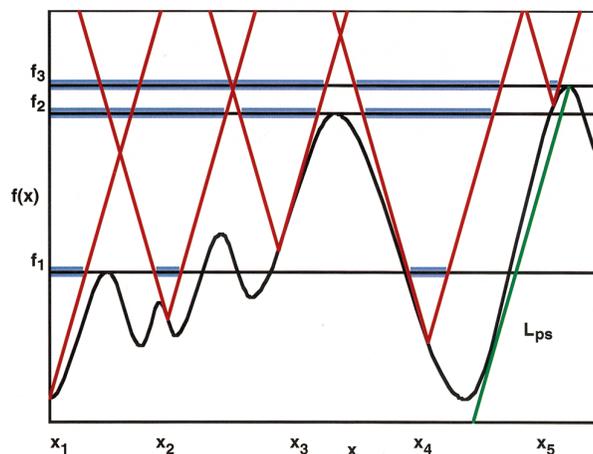


FIG. 1. Finding the global maximum for a 1-D example using the SPT algorithm.

parameter vector (\mathbf{x}) is much more expensive. During the ascent phase of the algorithm, derivatives are required to find a local maximum. While finding a local maximum is relatively expensive, the benefit is large. When we perform a search and find a higher value than the previous local maximum, we know that we are in the hill of attraction of a new local maximum that will be higher than the previous local maximum. Thus, we continually ascend toward the global maximum. An algorithm that ascends from many random starting points can repeatedly ascend into a previously identified local maximum. By searching for a higher value than the previous local maximum, we gain two benefits: We do not need to calculate derivatives while searching, and we calculate derivatives only when they will guide us to a new and higher local maximum.

The key parameter in the SPT algorithm is a pseudo-Lipschitz constant (L_{ps}) that is used to define unpromising regions in the search space that do not need to be explored. For the 1-D case (where the parameter vector is a scalar), the Lipschitz constant (L) is an upper bound on the rate of change of the objective function [$f(x)$]:

$$\left| \frac{df}{dx} \right| \leq L. \quad (18)$$

The pseudo-Lipschitz constant (L_{ps}) is the largest slope of any line drawn from the global maximum that is tangent to the curve that defines the hill of attraction for the global maximum (see Figure 1). In general, the pseudo-Lipschitz constant is less than the Lipschitz constant. For a square well (a function that is constant before point a increases to a higher value at point a and remains at the higher value until point b , where it returns to the original value), the Lipschitz constant is infinity while the pseudo-Lipschitz constant is finite (if we assume that the location of the global maximum is the midpoint between a and b).

We use Figure 1 to discuss a simple version of the SPT algorithm for the 1-D case. We evaluate the function at point 1 (x_1). From x_1 , we ascend to the local maximum (f_1). The local maximum is our best candidate for the global maximum: $f^* = f_1$. From x_1 , we draw a line (half of a Pijavskij cone) that has the same slope as the pseudo-Lipschitz line. The intersection of the line from x_1 and the line ($y = f_1$) determines a small region of the x -axis (the left shaded region on line f_1 in Figure 1) that cannot have a higher value than the current maximum value (f_1).

Rather than performing cycles of ascent and shooting, we randomly choose a second point in the available portion of the x -axis (x_2) and evaluate the function. If $f(x) < f^*$, we are shooting; if $f(x) > f^*$, we ascend to a local maximum. Since $f(x_2) < f^*$, we are shooting. From x_2 , we draw a Pijavskij cone (the magnitudes of the slopes of the two lines are equal to the slope for the pseudo-Lipschitz line). The intersection of the cone from x_2 and line f_1 excludes a small region of the x -axis.

We randomly choose a third point in the available portion of the x -axis (x_3) and evaluate the function. Since $f(x_3) > f^*$, we are in the hill of attraction of a new local maximum. From x_3 , we ascend to the local maximum (f_2) and set $f^* = f_2$. From x_3 , we draw a Pijavskij cone. Since the current local maximum is much higher than f_1 , the three Pijavskij cones now exclude much more of the x -axis. We choose a fourth point in the available portion of the x -axis (x_4) and evaluate the function. Since

$f(x_4) < f^*$, we are shooting. From x_4 , we draw a Pijavskij cone that excludes a large region of the x -axis. We choose a fifth point in the available portion of the x -axis (x_5) and evaluate the function. Since $f(x_5) > f^*$, we are in the hill of attraction of a new local maximum. From x_5 , we ascend to the local maximum (f_3) and set $f^* = f_3$. We evaluate the function at a few more points and conclude that f_3 is the global maximum for the example in Figure 1.

The Pijavskij cones exclude more of the x -axis as f^* increases. We can improve the efficiency of the SPT algorithm by adding a second parameter: a close upper bound for the global maximum (f^G) and using the maximum of f^* and f^G to define the cones. In the early stages of the algorithm, f^G will increase the region on the x -axis that is excluded by the Pijavskij cones. At the later stages of the algorithm, the current candidate for the global maximum, f^* , may become higher than f^G and the parameter will have no impact. After several attempts at finding the global maximum, we can use the best value found to date as an estimate of f^G .

We define the resolution as the width of the smallest hill of attraction that can be detected. For the searches that we discuss in the results section, we assume the range of each component of \mathbf{x} is ± 50 ms (for a total range of 100 ms). Since we use a resolution that allows 100 000 points in the range, our resolution is $1 \mu\text{s}$. For a comparison, Rothman (1986) had a range of ± 160 ms and a resolution of 8 ms, while DuBose (1993) had a range of ± 50 ms and a resolution of 2 ms. While our resolution is apparently more than three orders of magnitude tighter than the previous work, there is no physical justification for having a resolution that is greater than a few samples within one cycle of the highest frequency in the data set. If the highest frequency is 60 Hz, 2 ms is a reasonable resolution. The SPT method allows us to work at a much higher resolution than can be justified by the experimental details without a significant increase in computational cost. In the worst case there is a high penalty for high resolution. We can imagine an objective function that is constant except for randomly distributed square wells of varying height that are $1 \mu\text{s}$ wide. For this function, the Pijavskij cones would not eliminate any regions of the x -axis, and we would need to evaluate the objective function 100 000 times to find the global maximum. The stack energy problem is much better than the worst case: the Pijavskij cones quickly eliminate all of the x -axis, and we usually evaluate the function about 10 times to achieve a resolution of 100 000 points in the parameter range. Thus, we do not pay a penalty for working at high resolution.

We can use two complementary methods to estimate the pseudo-Lipschitz constant (L_{ps}): measure derivatives and set resolution. As we perform searches and descents, we calculate the function and the derivatives of the function with respect to each component of \mathbf{x} . We can monitor the derivatives and determine the largest values. The pseudo-Lipschitz constant should be larger than any measured value.

To use the resolution, we consider a 1-D search. The pseudo-Lipschitz constant is the ratio of a change in $f(x)$ to a change in x [see equation (18)]. The change in f is the difference between the current value of f and our close upper bound for the global maximum (f^G), while the change in x is one-half the resolution. Thus, our second parameter (f^G) and the resolution can be used to estimate the first parameter (the pseudo-Lipschitz constant). Since our resolution is very small, we calculate much

higher values for the pseudo-Lipschitz constant than we obtain by measuring derivatives.

We will describe the SPT algorithm for the general case where \mathbf{x} is an N -dimensional vector.

- 1) Select a starting point. The values of the components of \mathbf{x} could be all zero, read from a file, or random numbers. The input file could be the inverse map for the disaggregated problems [from equation (15)].
- 2) Use an ascent algorithm to find a local maximum.
- 3) Begin the outer loop.
- 4) Begin a loop over the N -components of \mathbf{x} . For each loop, randomly choose an integer (n) in the range 1 to N .
- 5) Evaluate a fixed number of random points x_{nr} for x_n . If x_{nr} is in the available portion of x_n , evaluate the function. Save the value for x_n that gives the highest value for $f(x)$.
- 6) If the search finds a point on a new hill of attraction, ascend to the next local maximum.
- 7) End of the loop that began in step 4.
- 8) End of the outer loop.

APPLICATIONS

We have applied our global optimization method to a synthetic data set that was created by adding disrupting statics to measured seismic data. The data set has 316 parameters (100 shots and 216 receivers), 4776 traces, and 423 CMPs. Using 118 Fourier components for the data, the stack energy at the initial point ($\mathbf{x} = \mathbf{0}$) was 882 and the upper bound on the stack energy [G , see equation (9)] was $G = \sum_k G_k = 6589$.

We applied our global optimization method to the 423 CMP stack energy problems. Most of the best values for the coherence factor were much less than one. The total energy for the 423 CMP problems (${}^D G$) was 2706. Using equation (15), the 4776 \mathbf{v} -coordinates were mapped back to the 316 \mathbf{x} -coordinates to determine an initial point (\mathbf{x}_d) for the 316-parameter global optimization problem. The initial value for the energy was 1035. The first local maximum was 2183. Using our SPT method, we found many points with high values of stack energy.

In this section, we compare the seismic image we were given with the image after applying residual statics corrections. We discuss both the coherence and convergence factors for our best solution and compare our best estimates for the statics with the disrupting statics used to create the data sets.

We begin with three seismic images: before disruption, disrupted, and after correction. The original seismic image before applying disrupting statics is displayed in Figure 2. The disrupted seismic image is plotted in Figure 3, while the image after the static corrections is shown in Figure 4. The seismic reflections are well defined in Figures 2 and 4 and are not defined in Figure 3.

Comparison of the original (Figure 2) image with the SPT (Figure 4) image demonstrates that both methods for calculating statics yield similar results. The reflections sequence at 1.7–1.8 s differs in at least two regards on the two images: (1) the reflections have slightly higher amplitude and perhaps more lateral continuity on the SPT image and (2) there is a marked difference in the stacked images between CMPs 100 and 200. In the original image, the reflection at about 1.76 s is continuous across the section. Reflections above and below the 1.76-s reflection have lower amplitude and are less continuous. In the SPT section, there is an apparent offset in the 1.76-s reflection

at about CMP 160. The offset appears to propagate upward and downward in the section, with a moderately steep dip toward the right side of the image. Reflections above 1.76 s appear to be more continuous in the SPT image than in the original image. This is also true of reflections between 1.76 and 1.85 s, but both images show more complicated structures at greater depths. At 1.9–2.0 s between CMPs 50 and 200, the images again differ significantly from one another. Here, the original image appears to indicate two closed cells, while the SPT image shows only one of these (centered at CMP 100, 1.94 s). In this portion of the record, it is difficult to select the better image.

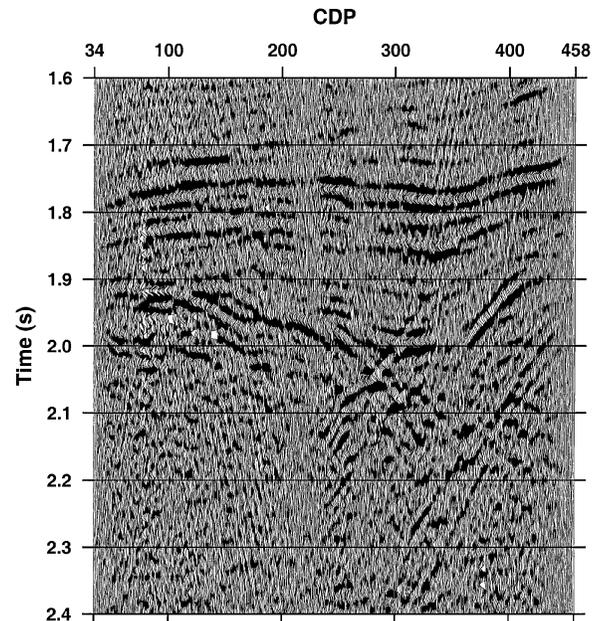


FIG. 2. The seismic image before the application of the disrupting statics.

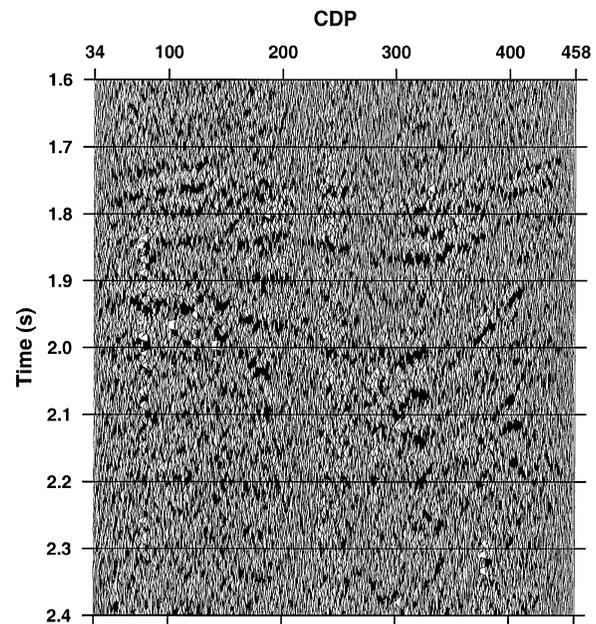


FIG. 3. The disrupted seismic image before applying the residual statics corrections.

Recall that when all of the traces in a stack are in phase, the coherence factor ($Q_k = E_k/G_k$) is one. The best value that we have found for the stack energy is 2441. Since the upper bound on the stack energy is 6589, the weighted average value for the coherence factors is 0.370. Thus, most of the traces are not in phase. We have solved the global optimization problems to find the maximum stack energy for each CMP: ${}^D G_k$. Using ${}^D G_k$, we can define a best (close upper bound for the) coherence factor (see Figure 5). Most of the values for the best coherence factor range from 30–50%.

The convergence factor compares our current values for the stack energy to the maximum of the CMP stack energy. In Figure 6, the convergence factor is plotted for two values of the stack energy: 882 and 2441. (Recall that 882 is the stack energy when all of the static corrections are zero.) Most of the values for the convergence factor are above 80% when the energy is large (2441).

After developing the SPT version of TRUST, we found many vectors with high values of stack energy. In Reister et al. (1999), we use both a modified Euclidean distance norm and an energy norm to measure the differences between a few (22) of the

vectors (potential solutions) with stack energy greater than 2365, and we conclude that the 22 points are distinct.

After we solved the problem, we compared our estimates of the statics with the disrupting statics that had been applied to the original seismic data (Figure 2) to produce the input data for this project (Figure 3). The disrupting statics are plotted in Figure 7 and range from -21 ms to 24 ms. Figure 8 displays the difference between our estimates of the disrupting statics for case 2441 and the applied disrupting statics [we plot the

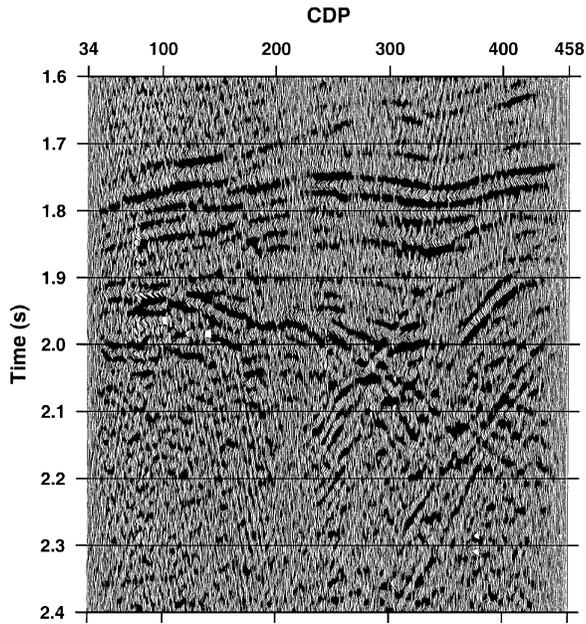


FIG. 4. The seismic image after applying the residual statics corrections.

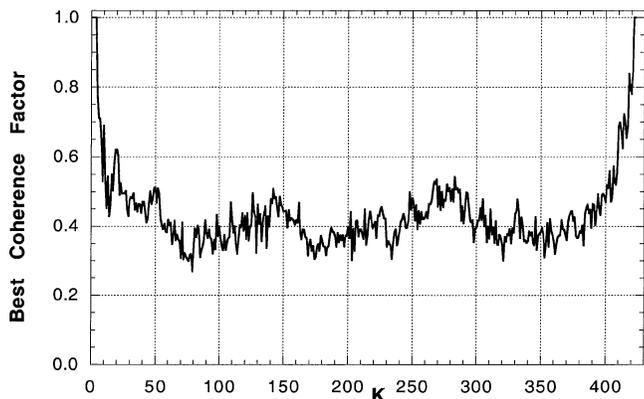


FIG. 5. The best coherence factor.

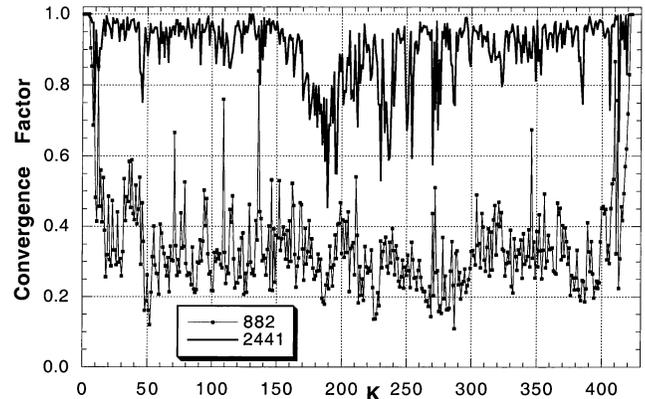


FIG. 6. The convergence factor for two cases: no corrections (882) and best corrections (2441).

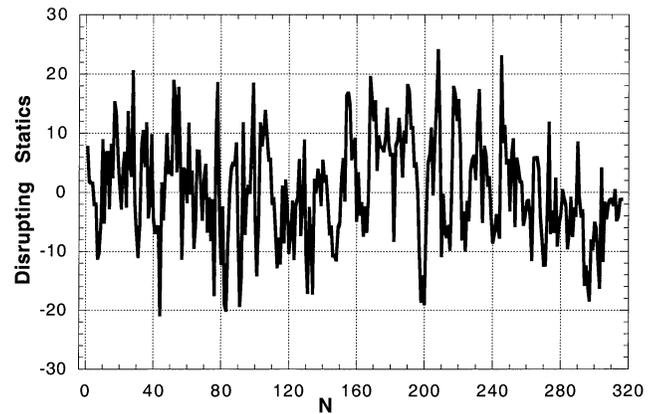


FIG. 7. The disrupting statics that were applied to the original seismic data (Figure 2) to produce the input data for this project (Figure 3). The units are in milliseconds.

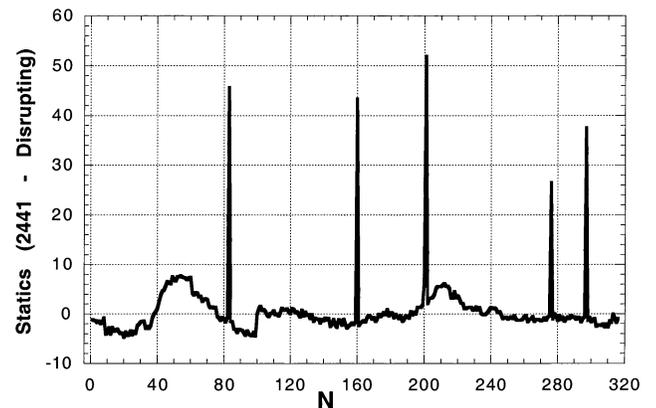


FIG. 8. The difference between the statics for the 2441 case and the disrupting statics. The units are in milliseconds.

differences for case 2427 in Reister et al. (1999)]. The most striking result is that five components of \mathbf{x} have large differences (cycle skips) for the two cases and the cycle skips occur for the same components (83, 160, 201, 276, and 297). When we plot the difference between case 2441 and case 2427 (in Reister et al., 1999), we do not find cycle skips.

When we calculated the stack energy for the applied disrupting statics in Figure 7, we expected to find a very large value for the energy and were surprised to find that the energy is 2349, which is lower than for the 22 cases. When we start from the applied disrupting statics and climb to the top of the first peak, we reach an energy of 2415 (better than all but three of our 22 cases). The differences between the statics for the 2415 case and the disrupting statics are plotted in Figure 9. There are no cycle skips. While the differences are smaller than the differences between cases 2441 and 2427, the differences are significant and range from -4 to 8 (thus, the differences are not from step size or round-off). Many methods can be used to estimate static corrections. Clearly, the static corrections for the original seismic image were not estimated by maximizing stack energy.

CONCLUSIONS

We have developed a new method for global optimization (SPT) designed to maximize stack energy. In general, we cannot solve a global optimization problem. We must apply resource allocation—apply finite resources (function evaluations) in an efficient way to find the best estimate of the global optimum. Beginning at an initial point, the SPT method ascends to the first local maximum. The method finds candidates for the global maximum by executing a series of cycles of search and ascent. At the start of each search cycle, we randomly choose one of the N parameters to be a variable. We evaluate a fixed number of random values for the variable. We use Pijavskij cones drawn from points where we have calculated the stack energy to restrict possible values for the variable. If the random value is in the acceptable region, we evaluate the stack energy. If (at the end of the evaluation) the highest value for the stack energy is higher than the current local maximum, we ascend to a new local maximum. Finally, we begin a new search cycle. At the end of the cycles, we would have a new candidate for the global maximum.

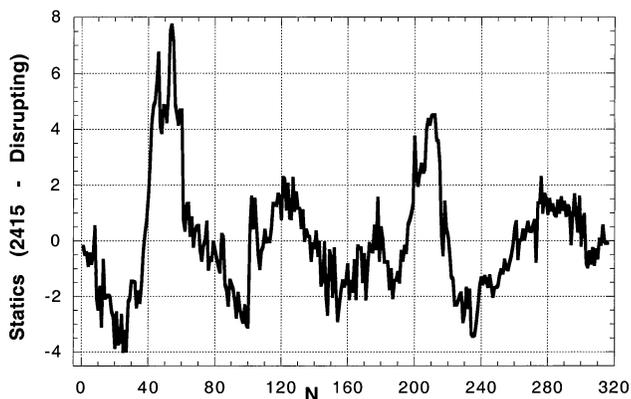


FIG. 9. The difference between the statics for the 2415 case and the disrupting statics. The units are in milliseconds.

The key parameter in SPT is the pseudo-Lipschitz constant, which determines the slope of the two sides of the Pijavskij cones that are used to exclude regions of the search space. The SPT method has two other parameters (an estimate of the global maximum and the resolution) that can be used to estimate the pseudo-Lipschitz constant. The Pijavskij cones quickly eliminate most of the search space, and we usually need to evaluate the function about ten times to exclude all 100 000 of the possible points in the parameter range.

To estimate a close upper bound for the total stack energy, we sum the best values of the CMP stack energy. The upper bound helps us estimate the global maximum (f^G) for the SPT method and provides a yardstick to measure our progress toward the global maximum. Since our best estimate (2441) is near the upper bound (2706) and the best of a large family of vectors with high stack energy, we believe we have found a good estimate for the global maximum. The solution of the CMP problems also provides a good initial estimate of the static corrections.

ACKNOWLEDGMENTS

This work was sponsored by both the Office of Fossil Energy (Office of Natural Gas and Petroleum Technology) and the Office of Science (Office of Basic Energy Sciences and Office of Advanced Scientific Computing Research), U.S. Department of Energy (DOE). Oak Ridge National Laboratory is managed for DOE by UT-Battelle, LLC under contract DE-AC05-00OR 22725. The reviewers provided many constructive suggestions on ways to sharpen the focus of the paper and make the ideas understandable to a general audience. W. E. Doll prepared the three seismic images and commented on the differences between the original image and the SPT image.

REFERENCES

- Barhen, J., and Protopopescu, V., 1996, Generalized TRUST algorithms for global optimization, *in* Floudas, C. A., and Pardalos, P. M., Eds., *State of the art in global optimization: Computational methods and applications*: Kluwer Academic Press, 163–180.
- Barhen, J., Protopopescu V., and Reister, D. B., 1997, TRUST: A deterministic algorithm for global optimization: *Science*, **276**, 1094–1097.
- Cetin, B. C., Barhen, J., and Burdick, J. W., 1993, Terminal repeller unconstrained subenergy tunneling (TRUST) for fast global optimization: *J. Optim. Theory Appl.*, **77**, 97–126.
- DuBose, J. B., Jr., 1993, Practical steps toward realizing the potential of Monte Carlo automatic statics: *Geophysics*, **58**, 399–407.
- Marsden, D., 1993, Static corrections—A review: *The Leading Edge*, **12**, 43–49, 115–120, 210–216.
- Oblo, E. M., 1999, SPT: A stochastic tunneling algorithm for global optimization: Oak Ridge National Laboratory publication ORNL/TM-1339, available from National Technical Information Service.
- Reister, D. B., Oblo, E. M., Barhen, J., and DuBose, J. B., Jr., 1999, Application of global optimization to the estimation of surface-consistent residual statics: Oak Ridge National Laboratory publication ORNL/TM-1999/102, available from National Technical Information Service.
- Ronen, J., and Claerbout, J. F., 1985, Surface-consistent residual statics estimation by stack-energy maximization: *Geophysics*, **50**, 2759–2767.
- Rothman, D. H., 1985, Nonlinear inversion, statistical mechanics, and residual statics corrections: *Geophysics*, **50**, 2784–2796.
- , 1986, Automatic estimation of large residual statics corrections: *Geophysics*, **51**, 332–346.
- Sheriff, R. E., and Geldart, L. P., 1995, *Exploration seismology*, 2nd Ed.: Cambridge Univ. Press.
- Taner, M. T., Koehler, F., and Alhaili, K. A., 1974, Estimation and correction of near-surface time anomalies: *Geophysics*, **39**, 441–463.
- Wiggins, R. A., Lerner, K. L., and Wisecup, R. D., 1976, Residual statics analysis as a general linear inverse problem: *Geophysics*, **41**, 922–938.
- Yilmaz, O., 1987, *Seismic data processing*: Soc. Expl. Geophys.