Tests of relative earthquake location techniques using synthetic data

Guoqing Lin and Peter Shearer

Institute of Geophysics and Planetary Physics, Scripps Institution of Oceanography, University of California, San Diego, La Jolla, California, USA

Received 12 August 2004; revised 18 January 2005; accepted 31 January 2005; published 8 April 2005.

[1] We compare three relative earthquake location techniques using tests on synthetic data that simulate many of the statistical properties of real travel time data. The methods are (1) the hypocentroidal decomposition method of Jordan and Sverdrup (1981), (2) the source-specific station term method (SSST) of Richards-Dinger and Shearer (2000), and (3) the modified double-difference method (DD) of Waldhauser and Ellsworth (2000). We generate a set of synthetic earthquakes, stations, and arrival time picks in half-space velocity models. We simulate the effect of travel time variations caused by random picking errors, station terms, and general three-dimensional velocity structure. We implement the algorithms with a common linearized approach and solve the systems using a conjugate gradient method. We constrain the mean location shift to be zero for the hypocentroidal decomposition and double-difference locations. For a single compact cluster of events, these three methods yield very similar improvements in relative location accuracy. For distributed seismicity, the DD and SSST algorithms both provide improved relative locations of comparable accuracy. We also present a new location technique, termed the shrinking box SSST method, which provides some improvement in absolute location accuracy compared to the SSST method. In our implementation of these algorithms, the SSST method runs significantly faster than the DD method.

Citation: Lin, G., and P. Shearer (2005), Tests of relative earthquake location techniques using synthetic data, *J. Geophys. Res.*, *110*, B04304, doi:10.1029/2004JB003380.

1. Introduction

[2] The classic problem of locating earthquakes using arrival time data has recently been revitalized by methods that can greatly improve the relative location accuracy among nearby events, even when the arrival times are biased by the effects of three-dimensional velocity structure. These new methods include the double-difference algorithm [*Waldhauser and Ellsworth*, 2000, 2002] and source-specific station terms [*Richards-Dinger and Shearer*, 2000]. Both of these techniques represent generalizations to distributed seismicity of methods that have previously been applied to isolated clusters of events, such as joint epicenter determination, station terms, and master event locations [e.g., *Douglas*, 1967; *Evernden*, 1969; *Lilwall and Douglas*, 1970; *Frohlich*, 1979; *Jordan and Sverdrup*, 1981; *Smith*, 1982; *Pavlis and Booker*, 1983; *Viret et al.*, 1984; *Pujol*, 1988]. All of these methods attempt to correct for the systematic biases in arrival times caused by three-dimensional velocity variations without actually solving for the velocity structure itself. They are effective in reducing the relative errors among nearby events (for which the arrival time perturbations are correlated) but typically do not significantly improve absolute location accuracy (which requires knowledge of the true three-dimensional velocity structure).

[3] Improvements in relative location accuracy obtained using these methods often produce a dramatic sharpening of seismicity patterns, particularly when more accurate timing is obtained using waveform cross correlation [e.g., *Rubin et al.*, 1999; *Waldhauser et al.*, 1999; *Waldhauser and Ellsworth*, 2000; *Shearer*, 2002]. Evaluating the performance of these methods is complicated, however, by the fact that the true earthquake locations are unknown. Here we generate synthetic data sets to compare four different relative earthquake location techniques: (1) the hypocentroidal decomposition method of *Jordan and Sverdrup* [1981], (2) the station term method [e.g., *Frohlich*, 1979], (3) the source-specific station term method (SSST) of *Richards-Dinger and Shearer* [2000], and (4) the modified double-difference method (DD) of *Waldhauser and Ellsworth* [2000]. We also introduce a variation on the SSST method, which we term "shrinking box" SSST, that has some advantages over the conventional SSST method.

[4] Our comparisons in this study are restricted to arrival time data alone, i.e., we do not consider waveform cross-correlation constraints. Our numerical experiments show that all of the methods give very close to the same result for the relative locations among nearby events, as should be expected from the theory underlying the methods. However, there are significant differences in the computational effi-

0148-0227/05/2004JB003380$09.00

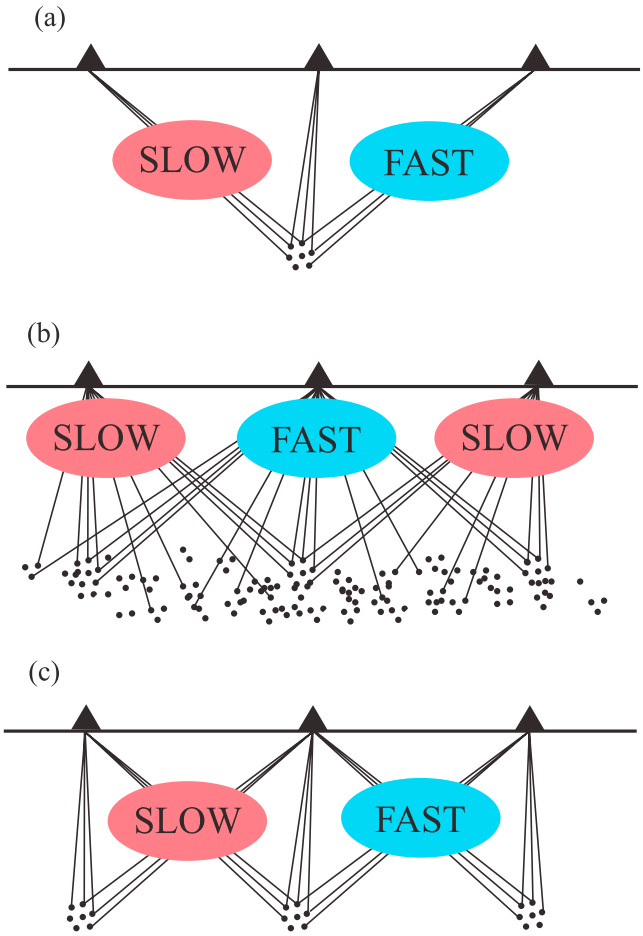

Figure 1. Cartoons illustrating how travel times and station terms are affected by three-dimensional velocity structure and different source-receiver geometries. (a) A compact cluster of events. The ray paths to each station pass through approximately the same velocity structure. In this case, a single travel time correction term at each station (static station terms) can account for the biasing effects of the three-dimensional structure. (b) Distributed seismicity. Static station terms cannot fully account for general 3-D structure. They may, however, provide an estimate of the biasing effect of the shallow velocity anomalies below each of the stations. This can lead to some improvement in the locations, particularly when the 3-D variations at depth are small compared to the near-surface variations. (c) Distributed seismicity in general 3-D structure. The travel time corrections to each station vary as a function of source position but are highly correlated among nearby events. Methods such as source-specific station terms (SSST) and double-difference (DD) can be used to improve locations in this case.

ciency of the methods for large data sets and the shrinking box method appears to have a slight advantage over the simple SSST method in terms of absolute location accuracy.

2. Linearized Earthquake Location

[5] Although the general earthquake location problem is nonlinear, the mathematics become much easier when we assume that perturbations to the locations are sufficiently small that a linear approximation is valid. In practical earthquake location algorithms this is typically achieved with an iterative approach, in which the location is assumed to be valid if the change in the location is very small at the final iteration.

[6] In this section, we review some earthquake location techniques following the notation by *Wolfe* [2002]. We begin with an introduction to the multiple-event location problem. Multiple-event location procedures are founded on the observation that the bias contaminating travel times from a set of nearby earthquakes tends to be strongly correlated; in particular, the error introduced by incorrect assumptions regarding Earth structure has a nearly constant value for arrival times measured at the same station (see Figure 1a). Since path bias of this type dominates the sample standard deviations computed for single-event locations [e.g., *Freedman*, 1967], the relative locations of events within a seismic cluster can be improved by taking these correlations into account.

[7] Suppose we have a set of $p = 1,2, .., P$ earthquakes, with each earthquake constrained by N_p arrival time observations. For simplicity we describe the situation in which P wave phases alone are used and the data are weighted equally, but the methods can be generalized to cases where S or other phases are also included and the arrival times are assigned different weights. Given an initial location estimate $\mathbf{x}_{p0}$ for an earthquake p, a linearized estimate for how the arrival time residuals respond to small changes in the location may be written

$$\mathbf{A}_p \, \Delta\mathbf{x}_p + \mathbf{s}_p = \Delta\mathbf{t}_p \tag{1}$$

where $\Delta\mathbf{t}_p$ is a N_p component vector containing the arrival time residuals, $\mathbf{A}_p$ defines the matrix of size $N_p \times 4$ containing the partial derivatives, calculated at the initial estimate $\mathbf{x}_{p0}$, $\Delta\mathbf{x}_p$ are the changes in earthquake hypocentral parameters (4×1) which we wish to determine (typically these are dx, dy, dz, dt), and $\mathbf{s}_p$ is a N_p component vector of the path anomalies due to velocity heterogeneity along each of the source-receiver ray paths. The new locations will be updated as $\mathbf{x}_p = \mathbf{x}_{p0} + \Delta\mathbf{x}_p$. When equation (1) is applied to locate a single earthquake, $\mathbf{s}_p$ is either set to zero or to predetermined values. We will term this approach "single-event location," meaning that no information is used from any other events.

[8] In most cases we compute the partial derivatives with respect to a one-dimensional reference seismic velocity model, although this is not required by the method. The path correction terms $\mathbf{s}_p$ then represent the biasing effects of the unmodeled three-dimensional structure or other errors in the velocity model.

[9] Considering all P earthquakes as one linearized system, we may combine the individual single event location equations (1) into

$$\mathbf{A}\,\Delta\,\mathbf{X} + \mathbf{S} = \Delta\mathbf{T} \tag{2}$$

in which the locations are updated as $\mathbf{X} = \mathbf{X}_0 + \Delta\mathbf{X}$. In this case, we define M_T as the total number of unknown location parameters ($4P$) and N_T as the total number of arrival time observations. $\Delta\mathbf{X}$ is an $M_T \times 1$ vector in which the

individual $\Delta\mathbf{x}_p$ vectors are strung end to end, and $\mathbf{S}$ and $\Delta\mathbf{T}$ are $N_T \times 1$ vectors with $\mathbf{s}_p$ and $\Delta\mathbf{t}_p$ strung end to end, respectively. $\mathbf{A}$ is an $N_T \times M_T$ matrix, containing the individual partial derivative $\mathbf{A}_p$ matrices along its diagonal, i.e.,

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & . & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & . & \mathbf{0} \\ . & . & & . \\ . & . & & . \\ \mathbf{0} & . & . & \mathbf{A}_P \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \\ . \\ \Delta\mathbf{x}_P \end{bmatrix} + \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ . \\ . \\ \mathbf{s}_P \end{bmatrix} = \begin{bmatrix} \Delta\mathbf{t}_1 \\ \Delta\mathbf{t}_2 \\ . \\ . \\ \Delta\mathbf{t}_P \end{bmatrix}$$

As written, this equation permits the trivial solution $\Delta\mathbf{X} = \mathbf{0}$ and $\mathbf{S} = \Delta\mathbf{T}$, in which the individual path anomaly correction terms are set to the travel time residuals for each source-receiver ray path. Meaningful solutions are only possible if we apply constraints to the correction terms to reduce the number of free parameters. These constraints typically assume that the correction terms are correlated for nearby ray paths. The simplest form of these constraints is to assume that the earthquakes are in a compact enough cluster that the path anomaly to each seismic station is constant.

[10] In this case if K_T is the number of stations providing observations for the set of events, then the full correction terms vector $\mathbf{S}$ can be expressed as

$$\mathbf{S} = \mathbf{Bs} \tag{3}$$

where $\mathbf{s}$ is a $K_T \times 1$ vector that contains the terms at each station and $\mathbf{B}$ is the $N_T \times K_T$ matrix that assigns one of these K_T terms to each of the N_T travel time residuals:

$$\mathbf{B}_{ij} = \begin{cases} 1 & \text{when } \Delta\mathbf{T}_i \text{ is from station } j \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

and (2) becomes

$$\mathbf{A}\,\Delta\mathbf{X} + \mathbf{Bs} = \Delta\mathbf{T} \tag{5}$$

[11] We will term the $\mathbf{s}$ values the "static" station terms to indicate that they have fixed values for each station regardless of the event location. In contrast, the "source-specific" station terms (SSST) that we will later consider have different values for different source locations. It is important to note at this point that (5) does not have a unique least squares solution for $\Delta\mathbf{X}$ and $\mathbf{s}$ because their projections onto the data space are not linearly independent. The most obvious nonuniqueness is the tradeoff between the event origin time part of $\Delta\mathbf{X}$ and the station terms; any constant time could be added to one and subtracted from the other. This ambiguity can be removed fairly easily by imposing additional constraints on $\mathbf{s}$, for example, by forcing the mean station term to zero.

[12] However, there are also tradeoffs between the x, y, z locations of the events and the values of $\mathbf{s}$. For example, north-south shifts in the absolute event locations can be accommodated by adding times to station terms to the north of the event cluster and subtracting times from the station terms to the south. This tradeoff between the station terms and the locations is complete if the partial derivatives are identical for all of the events (i.e., the same reference location $\mathbf{x}_{p0}$ is used) [*Jordan and Sverdrup*, 1981; *Pavlis and Hokanson*, 1985]. In this case the absolute location of the event cluster is unconstrained, provided no additional constraints are imposed on $\mathbf{s}$. When the partial derivative matrices differ among the events, in principle this tradeoff is broken and absolute location information can be obtained. However, in practice the system remains very ill conditioned and absolute locations are not significantly constrained until the event separation becomes quite large (at which point the static station term approximation is probably no longer valid because the path anomalies to each station will vary between events).

[13] The improvement in location accuracy obtained by using (5) is found in the relative locations among the events in the cluster. One might ask why this is achieved given that the path anomalies are the same for all of the events and could be expected to have a similar biasing effect on all of the event locations. The improvement occurs largely because of the fact that arrival times for the events are not always recorded by the same set of stations and thus are biased by the path anomalies by differing amounts. In addition, least squares solutions will tend to weight the largest residuals the most and thus the solutions may be dominated by the small set of stations with the largest station terms.

[14] In the following sections, we will show how different techniques improve relative locations.

2.1. Hypocentroidal Decomposition (JS)

[15] This method was introduced by *Jordan and Sverdrup* [1981] (hereinafter referred to as JS) and involves projecting out the part of the problem that is sensitive to the static station terms. In this case we define the $N_T \times N_T$ projection operator $\mathbf{Q_{JS}}$:

$$\mathbf{Q_{JS}} \equiv \mathbf{I}^{N_T} - \mathbf{BB}^{\dagger} \tag{6}$$

where $\mathbf{B}^{\dagger}$ is the generalized inverse of $\mathbf{B}$ calculated by singular value decomposition (SVD).

[16] Define w_k to be the number of earthquakes recorded by station k and $w_{k(i)}$ to be the number of earthquakes recorded by the station recording $[\Delta\mathbf{T}]_i$. Then

$$[\mathbf{Q_{JS}}]_{ij} = \delta_{ij} - \frac{1}{w_{k(i)}}\Delta_{ij} \tag{7}$$

where δ_{ij} is the Kronecker delta function and $\Delta_{ij} = 1$ if $[\Delta\mathbf{T}]_i$ and $[\Delta\mathbf{T}]_j$ are from same station, and zero otherwise.

[17] $\mathbf{Q_{JS}}$ transforms $\Delta\mathbf{T}$ into $\Delta\mathbf{T}$ minus the travel time residual averages at each station. In the static station term case, all of the terms are equal to these averages, and thus $\mathbf{Q_{JS}B} = \mathbf{0}$ and (5) becomes

$$\mathbf{Q_{JS}A}\,\Delta\mathbf{X} = \mathbf{Q_{JS}}\Delta\mathbf{T} \tag{8}$$

The least squares solution to this equation can be obtained as

$$\mathbf{\Delta X} = (\mathbf{Q_{JS}A})^{\dagger}\mathbf{Q_{JS}\Delta T} \tag{9}$$

[18] Thus the new locations can be obtained without solving explicitly for the station terms. Note that stations that only record one earthquake will not contribute to this solution because the average residual in this case will always equal the residual itself; these stations can simply be deleted from the system prior to any calculation.

[19] Further insight into this method may be obtained by decomposing the set of location perturbations $\{\mathbf{\Delta x}_p;\ p = 1,2,\cdots,P\}$ into two parts:

$$\mathbf{\Delta x}_p = \mathbf{\Delta x}_0 + \mathbf{\delta x}_p \tag{10}$$

where

$$\mathbf{\Delta x}_0 \equiv P^{-1}\sum_{p=1}^{P}\mathbf{\Delta x}_p \tag{11}$$

and

$$\sum_{p=1}^{P}\mathbf{\delta x}_p = 0 \tag{12}$$

$\mathbf{x}_0 + \mathbf{\Delta x}_0$ is called the *hypocentroid*, the average absolute location of the earthquakes, and the $\{\mathbf{\delta x}_p\}$ are called the *cluster vectors*, which only define the relative locations of the earthquakes, of the event group. Consequently,

$$\mathbf{\Delta X} = \mathbf{\Delta X}_0 + \mathbf{\delta X} \tag{13}$$

[20] The method of *Jordan and Sverdrup* [1981] was developed to study groups of shallow teleseismic earthquakes over limited regions. In this situation, the partial derivatives do not vary much with earthquake location, so in the Jordan and Sverdrup method, the partial derivatives in $\mathbf{A}$ for all earthquakes are set to an identical reference point. In this special case, it can be proved that $\mathbf{Q_{JS}}\ \mathbf{A\Delta X}_0 = \mathbf{0}$ [*Wolfe*, 2002], so that only improved relative locations can be obtained. In practice, events within a cluster are located first using single event location. The mean cluster location is then fixed as the reference location (the hypocentroid) for the cluster and the method solves for the perturbations to this location. For clusters with large numbers of events, considerable computation will be involved in the singular value decomposition of the $N_T \times M_T$ matrix $\mathbf{Q_{JS}A}\ \mathbf{\Delta X}$. In these cases, we have found that iterative matrix inversion techniques such as the conjugate gradient method are effective in speeding the calculations.

2.2. Static Station Terms (ST)

[21] One of the simplest and most widely applied relative location approaches is the station term method, which solves iteratively for a custom set of station-timing corrections [e.g., *Frohlich*, 1979; see also *Pujol*, 1988]. Equation (5), $\mathbf{A\Delta X} + \mathbf{Bs} = \mathbf{\Delta T}$, can be written

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{0} & . & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 & . & \mathbf{0} \\ . & . & & . \\ . & . & & . \\ \mathbf{0} & . & . & \mathbf{A}_P \end{bmatrix} \begin{bmatrix} \mathbf{\Delta x}_1 \\ \mathbf{\Delta x}_2 \\ . \\ \mathbf{\Delta x}_P \end{bmatrix} + \begin{bmatrix} 0 & 1 & . & 0 \\ 0 & 0 & . & 1 \\ & . & . & . \\ & . & . & . \\ 1 & 0 & . & 0 \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ . \\ \mathbf{s}_P \end{bmatrix} = \begin{bmatrix} \mathbf{\Delta t}_1 \\ \mathbf{\Delta t}_2 \\ . \\ . \\ \mathbf{\Delta t}_P \end{bmatrix} \tag{14}$$

[22] This is a coupled set of equations for the location parameters, $\mathbf{\Delta X}$, and the station terms, $\mathbf{s}$. We solve this iteratively by alternatively solving for each vector, while leaving the other vector fixed. In the first step the station corrections are held fixed (either set to zero or to values obtained elsewhere) and we solve for $\mathbf{\Delta X}$ using

$$\mathbf{A\,\Delta X} = \mathbf{\Delta T} - \mathbf{Bs} \tag{15}$$

The events are located with respect to a corrected set of arrival times. Because there is no coupling between the event locations at this step, each event can be located separately from the other events. Next, we solve for a new set of station terms using

$$\mathbf{Bs} = \mathbf{\Delta T} - \mathbf{A\,\Delta X} \tag{16}$$

Notice that the right hand side is simply the arrival time residuals; the least squares solution for $\mathbf{s}$ will set each station term to the mean residual of all of the events for the station. The process is then repeated until a stable set of locations and station terms is obtained. In Appendix A, we demonstrate that this algorithm should converge. In practice, we have found in most cases that convergence is rapid and that no more than 5 to 10 iterations are necessary.

[23] The method is much faster than hypocentroidal decomposition because there is no need to use the full $\mathbf{A}$ matrix in the calculations. The approach is also quite flexible because the station term calculation is performed separately from the event location calculation, so that any desired location method can be used, including standard or preexisting algorithms. However, the nonuniqueness inherent in these equations between the mean cluster location and

the station terms still exists. In practice, the mean cluster location is largely determined at the first iteration for the event locations.

[24] Finally, we note that the static station term method can be applied even when the events are not in a compact cluster (Figure 1b). In this case, the station terms will only be weakly correlated among different events at the same station because the source-receiver ray paths will sample different parts of the three-dimensional velocity structure. However, the method may nonetheless yield some improvement in location accuracy because differences in the shallow velocity structure beneath each station will be accounted for by the station terms. In this case, more accurate absolute event locations are possible, depending upon the details of the three-dimensional velocity structure that is biasing the times, azimuthal coverage and relative sizes of the velocity perturbations.

2.3. Source-Specific Station Terms (SSST)

[25] Static station terms work best when the differences between the actual travel times in the Earth and those in the assumed velocity model to each station are the same for all events. When the seismicity covers a large region containing significant lateral velocity heterogeneity (e.g., Figure 1c), neither the hypocentroidal decomposition method nor the static station term method is likely to work very well. In this case, a generalization of the station term approach, termed the source-specific station term (SSST) method by *Richards-Dinger and Shearer* [2000] can be applied. In this method, the location and station term calculations are again performed alternatively and the solution is obtained iteratively, but the station terms no longer consist of a single value at each station; rather, each station will have a station correction function which will vary as a function of source position.

[26] For static station terms, we simply calculate the station term for each station as the mean of the residuals at that station from all events. For source-specific station terms (SSSTs), we calculate a separate correction for each source-receiver pair at the given station using the residuals from nearby events. In this case, there is a different value of the station term vector $\mathbf{S}$ for every value of the residual vector $\mathbf{\Delta T}$; these values are a smoothed version of the event-specific residual field for each station. This smoothing over adjacent events can be done in a number of different ways. *Richards-Dinger and Shearer* [2000] smoothed over a fixed number of neighboring events using a natural neighbor tesselation.

[27] Here we will implement the SSST approach by selecting the nearby events that are located within a sphere of specified radius $r_{\max}$ around the target event. The station term for the target event is then computed as the mean residual of these events. Note that different results will be obtained depending upon the size of the cutoff distance. If $r_{\max}$ is set to a large enough distance, then the SSST method will give the same result as the static station term method. However, if $r_{\max}$ is set to a very small distance, the number of events may not be sufficient to obtain a reliable estimate of the true station term. Thus selection of the cutoff distance is a key factor in application of the SSST method.

[28] The SSST method shares some of the advantages of the static station term technique. The event location part of the calculation is separate from the station term calculation and can be performed quickly using any desired single-event location method. Computing the station terms at each iteration is also a simple calculation; the most numerically taxing part of this is identifying the events within the cutoff radius for each target event. In practice, convergence to a stable set of locations and station terms requires only a few iterations (although we have not derived a formal proof of convergence for the SSST algorithm). Note that in theory, the ST and SSST methods should yield the same locations for a single compact cluster that is smaller than the applied SSST cutoff distance $r_{\max}$.

[29] When *Richards-Dinger and Shearer* [2000] applied the SSST method to locate southern California earthquakes, they obtained their initial locations using the static station term method, and then used these station terms as a starting point for the SSST calculation. In this way, they achieved some improvement in the absolute locations of the events before focusing on the relative locations among closely spaced events. A generalization of this approach is to continuously shrink the SSST cutoff distance $r_{\max}$ between the first and final iteration. In other words, we start the cutoff distance with a large value to include all the events from which we calculate station terms, then decrease it to some specified minimum distance to calculate station terms using only the closest events. We will show later that this method seems to give the best results on our synthetic data sets.

2.4. Double-Difference (DD)

[30] The double-difference location algorithm [*Waldhauser and Ellsworth*, 2000, 2002] allows the simultaneous relocation of distributed events by minimizing residual differences for pairs of earthquakes without explicitly solving for station corrections.

[31] The single-event location problem for the kth observation of earthquake i can be written as

$$\frac{\partial t_k^i}{\partial \mathbf{x}} \mathbf{\Delta x}^i + s_k^i = \mathbf{\Delta} t_k^i \tag{17}$$

where s_k^i is the path anomaly correction between event i and station k. Thus we can express the time difference between the residuals at the same station for two events i and j as

$$\frac{\partial t_k^i}{\partial \mathbf{x}} \mathbf{\Delta} \mathrm{x}^i + s_k^i - \frac{\partial t_k^j}{\partial \mathbf{x}} \mathbf{\Delta} \mathrm{x}^j - s_k^j = dr_k^{ij} \tag{18}$$

where dr_k^{ij} is the residual between observed and calculated differential travel times between these two events, i.e.,

$$dr_k^{ij} = \left(t_k^i - t_k^j\right)^{\mathrm{obs}} - \left(t_k^i - t_k^j\right)^{\mathrm{cal}} \tag{19}$$

[32] If the events are close to each other, then their path corrections are likely to be similar, and so making the approximation $\mathbf{s}_k^i = \mathbf{s}_k^j$, the path anomalies cancel and we have simply

$$\frac{\partial t_k^i}{\partial \mathbf{x}} \mathbf{\Delta x}^i - \frac{\partial t_k^j}{\partial \mathbf{x}} \mathbf{\Delta x}^j = dr_k^{ij} \tag{20}$$

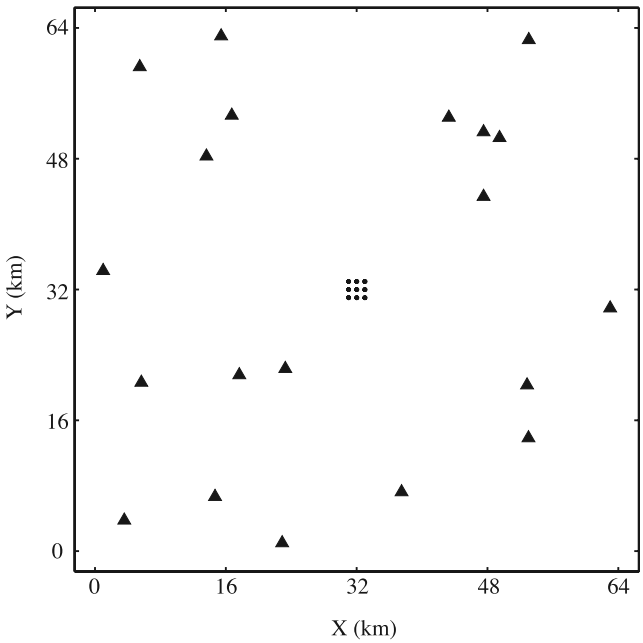

Figure 2. Map view of 20 random station locations (triangles) and 27 specified true event locations (dots) in a $2 \times 2 \times 2$ km grid at 10 km depth (each dot represents locations at 9, 10, and 11 km depth).

After combining these equations from all event pairs for a station, and for all stations, we obtain a system of linear equations of the form

$$\mathbf{G}\Delta\mathbf{X} = \mathbf{d} \tag{21}$$

where $\mathbf{G}$ defines a matrix of size $M_{DD} \times M_T$ (M_{DD} is the number of double-difference observations) containing partial derivatives, $\Delta\mathbf{X}$ is a vector containing the changes in hypocentral parameters that we wish to determine, and $\mathbf{d}$ is the data vector containing the double-difference times.

[33] Equivalently, this can be written as

$$\mathbf{Q_{DD}}\mathbf{A}\Delta\mathbf{X} = \mathbf{Q_{DD}}\Delta\mathbf{T} \tag{22}$$

where

$$\mathbf{Q_{DD}}\mathbf{A} = \mathbf{G} \tag{23}$$

and

$$\mathbf{Q_{DD}}\Delta\mathbf{T} = \mathbf{d} \tag{24}$$

[34] $\mathbf{Q_{DD}}$ is the double-difference operator combining differences of earthquake arrival times recorded at a given station k, $\Delta t_k^i - \Delta t_k^j$ (for $i \neq j$ and $i < j$) and has the form

$$\begin{bmatrix} 1 & . & . & -1 & . & . & . & . & 0 \\ 1 & . & . & . & . & . & -1 & . & . \\ . & . & . & . & . & . & . & . & . \\ 0 & . & . & . & . & 1 & . & . & -1 \end{bmatrix} \tag{25}$$

with each row containing only two nonzero terms, 1 and -1.

[35] The double-difference approach permits different types of differential arrival time data and choices in terms of selection and weighting of these data. One of the advantages of DD is that it is easy to incorporate *dt* results

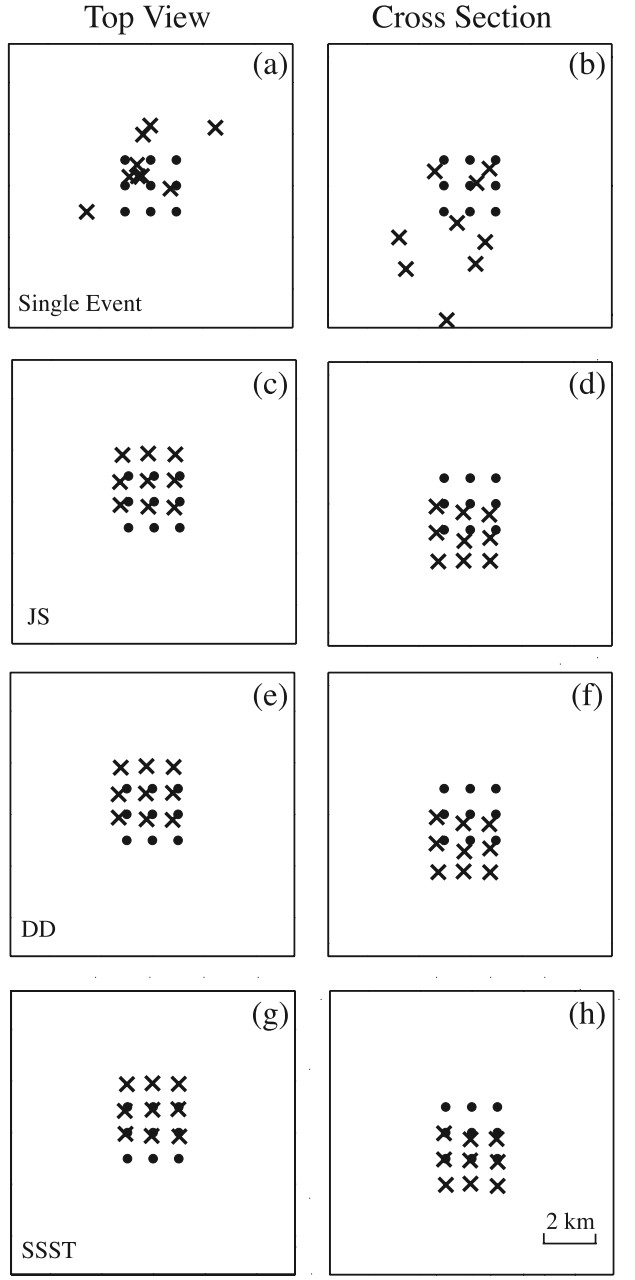

Figure 3. Comparison of different location methods for a single compact cluster of 27 events from one random realization. Dots are true locations, and crosses are computed locations. (a) Single event location top view. (b) Single event location cross section. (c) JS location top view. (d) JS location cross section. (e) DD location top view. (f) DD location cross section. (g) SSST location top view. (h) SSST location cross section. In each case, the middle (middle layer for top view and middle standing surface for cross section) nine events are shown, not the full 27.

Table 1. Single Cluster Location Results Comparison[a]

	RMS of Absolute Error		RMS of Relative Error	
Method	Horizontal, km	Vertical, km	Horizontal, km	Vertical, km
Single Event	1.55	2.26	1.14	2.41
JS[b]	1.10	1.54	0.07	0.34
DD[b]	1.10	1.54	0.07	0.34
SSST	0.91	1.69	0.06	0.36

[a]We show absolute and relative location RMS errors from 60 random realizations. For a given event, the relative locations are with respect to all the other events in the cluster.
[b]No mean location shift during relocation.

obtained from waveform cross correlation. DD can also be applied to phase data alone by selecting *dt* pairs from the individual arrival time picks. For small data sets consisting of a single isolated cluster it is practical to use all available event pairs. For large numbers of events or distributed seismicity, it makes sense to restrict the event pairs to those within some specified maximum separation distance. The algorithm released by *Waldhauser* [2001] allows the user to select this separation distance, as well as weight the differential times by distance and include absolute arrival time data if desired.

[36] To facilitate comparisons with the other methods, we wrote our own version of the double-difference algorithm. We do not use damping in any of the methods we test. Since the distance weighting of Waldhauser and Ellsworth should only work when damping is employed [*Wolfe*, 2002], we do not apply distance weighting in our location tests. We do, however, include an event separation distance cutoff. Thus $\mathbf{Q_{DD}}$ does not combine all possible differences of arrival times recorded by the same station, instead, we only consider the events in a sphere centered on the targeted event with the radius set to the cutoff distance, as discussed previously for the SSST algorithm. Our DD method uses only these differential times to refine the locations; the absolute pick times are used to obtain the initial locations but are not used in the DD algorithm itself.

3. Synthetic Data Tests

[37] We perform all our synthetic tests in a 64 × 64 × 32 km uniform half-space with a *P* wave velocity of 6 km/s and a *P*-to-*S* velocity ratio of 1.73. We generate 20 random station locations on the surface of the half-space and a set of specified earthquake locations. Although more realistic structures with slower near-surface velocities may result in downgoing ray paths from the source, our model provides reasonably approximate ray paths for the 9–11 km deep events that we model. To simulate the irregular pick availability of real earthquake data, for each event we generate *P* picks with 0.67 probability and *S* picks with 0.5 probability. Thus each synthetic event is recorded by a different set of stations with an average of 13 *P* picks and 10 *S* picks per event.

[38] The linear location system is solved by minimizing the L2 norm of the travel time residuals using a conjugate gradient algorithm. We compute 100 conjugate gradient iterations, although in most cases the solution converges much more quickly, depending on the data set and the starting conditions. For the SSST method, we used 100 conjugate gradient iterations at each location step, and found that 5 to 10 iterations for the station term calculation was sufficient. We begin all of the methods with linearized single event location to obtain a set of starting locations that are typically shifted somewhat from the true locations, depending upon the perturbations applied to the travel time picks. We will consider two forms of errors in the locations. Absolute location error is the difference between the computed location and the true location. Relative error is the difference between the computed and actual relative locations of two nearby events. Thus the concept of relative location error only makes sense for nearby events.

[39] We start our relocation with single event location, for which the starting locations are some random locations shifted from the true locations. Then the new locations from the single event location method are input as the starting locations for the three targeted techniques. The hypocentroidal decomposition method does not improve the mean location of the whole cluster, but the relocations significantly depend on the starting locations. We choose the mean location of the starting cluster as the common reference point and the partial derivatives are fixed to this point. We constrain both JS and DD methods to have no mean location shift relative to this reference point during relocation because we found that even a small amount of random picking error can produce unstable results for the DD algorithm if this constraint is not applied. It is likely that this constraint would be unnecessary if we used absolute pick times as well as relative times in our implementation of the DD method.

3.1. Single Isolated Cluster Tests

[40] In principle, all of the location techniques should yield similar relative locations for a compact event cluster. Our first test is to locate 27 events in a compact 2 × 2 × 2 km cube with the center at 10 km depth. We apply a cutoff distance for the DD and SSST algorithms that is large enough to cover all the events in the cube; in this case the static station term and SSST methods are equivalent. Figure 2 shows the distributions of the stations and events from one random realization of the station positions.

[41] We add two types of noise to the theoretical travel times: (1) Gaussian distributed station terms with zero mean and 0.3 s standard deviation (SD) for *P* picks (scaled by 1.73 for *S* picks), which are constants for all events recorded by a given station; (2) Gaussian distributed random picking errors with zero mean and 0.01 s standard deviation for *P* picks and 0.02 s for *S* picks. Random picking errors introduce location errors that cannot be improved with

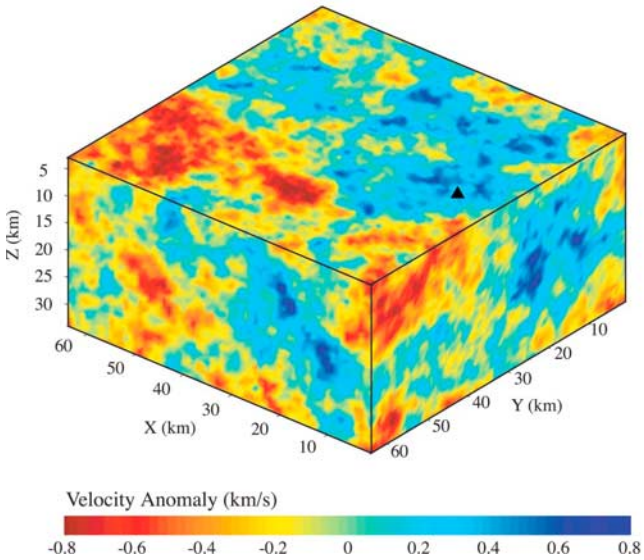

Figure 4. A random three-dimensional P velocity model relative to a constant velocity of 6 km/s, generated using a $k^{-1.8}$ power law.

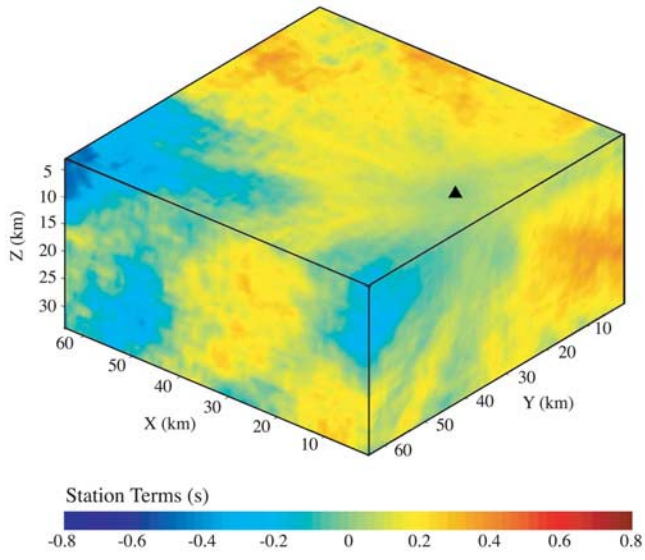

Figure 5. Source-specific station terms for the station shown as the triangle, corresponding to the random 3-D velocity model in Figure 4.

any of the methods discussed here. Since the focus of our study is on the relative location improvement that can be achieved by accounting for the station terms, we assign small values to the random pick errors in order to enhance the improvements that we will achieve. However, it is important to include at least some random picking error because otherwise the algorithms might achieve improvements in absolute location accuracy that are unlikely to be obtained with real data.

[42] Figure 3 shows the location results of one random realization of our synthetic data. As we expect, the single event location method produces quite large errors in both absolute and relative locations. These locations are equivalent to the standard (catalog) locations for real data. The shift in the mean cluster location (which dominates the absolute location errors for the JS, DD, and SSST methods), results from the random station terms and the finite number of stations. The size and direction of this shift vary among the difference random realizations of the synthetic data. The location map views and cross sections for this example clearly indicate that all three techniques, JS, DD, and SSST, yield similar improvements in relative location accuracy. Furthermore, JS and DD provide almost the same absolute locations, while SSST relocations are a little different from those of JS and DD. This is mainly due to the constraint on the JS and DD methods that the mean location shift is zero during relocation.

[43] We note that in principle, this constraint on the mean location shift is not required for the JS and DD methods and it might be possible in some circumstances to improve the absolute location accuracy by relaxing this requirement. In practice, however, we found that if even a very small amount of random picking error is present in the synthetic data, the absolute locations for the DD method sometimes become very unstable and unreliable unless the zero mean location shift constraint is applied.

[44] We estimate the relocation errors from 60 random realizations of the synthetic data. In Table 1, we show the root-mean-square (RMS) of both absolute and relative location errors. The relative location errors are calculated for each event relative to all the other events in the cube. As expected, all three methods (JS, DD and SSST) show improvement in location accuracy compared to the single event locations. Although some improvement is achieved in the absolute location accuracy (by reducing the scatter in the locations, not from any significant shift in the mean cluster

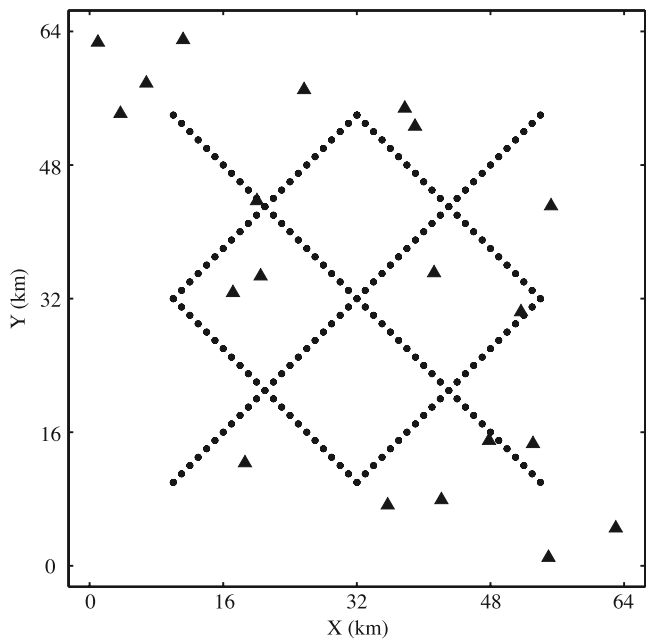

Figure 6. Earthquake locations and 20 random station locations for the distributed seismicity test of the location methods. Each dot represents earthquakes at 9, 10, and 11 km depth in this top view.

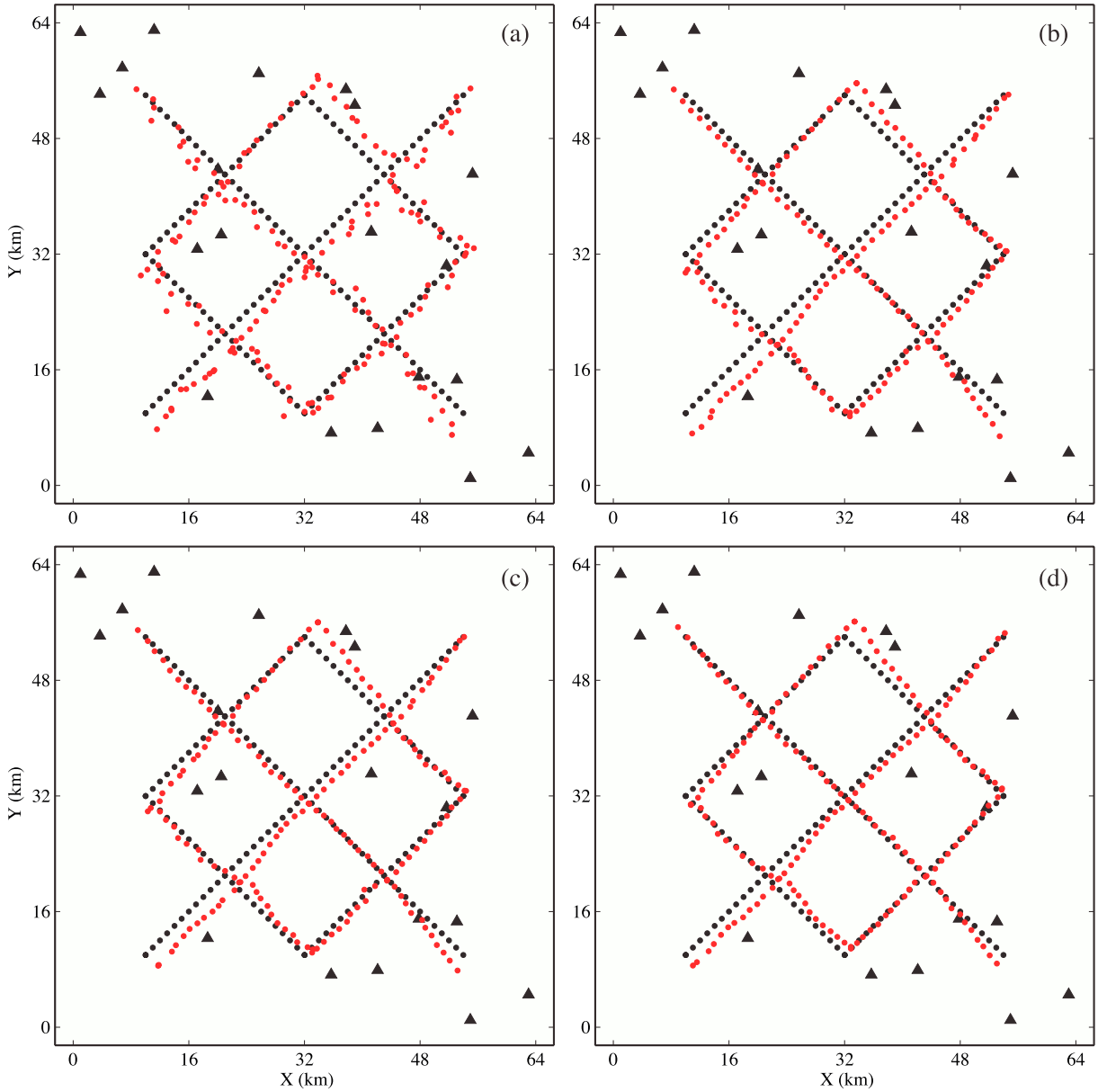

Figure 7. Comparison of different location methods for one random realization of the three-dimensional velocity structure shown in Figure 4. Black dots are true locations, and red dots are computed locations, and triangles are stations. (a) Single event location. (b) DD location. (c) SSST location. (d) Shrinking box SSST location. In each case, the middle 183 events are shown, not the full 549.

location), the most dramatic improvement is in the relative location accuracy, where the relative errors in horizontal position are reduced from 1.1 km to about 70 m. As is typically the case with real data, the horizontal location accuracy is much better than the vertical location accuracy.

3.2. Three-Dimensional Velocity Model Tests

[45] It is not surprising that all three techniques provide similar improvements in relative location accuracy for a compact cluster. The advantage of the SSST and DD methods is that they can also be applied to distributed seismicity. In this case, the events are far enough apart that the travel time perturbations cannot be assumed constant for each station. To generate realistic synthetic data for distributed events, we compute travel time differences resulting from an isotropic three-dimensional velocity model with random P velocity variations of 0.3 km/s RMS using a $k^{-1.8}$ power law. S velocity variations are scaled as 3 times the P variations. We sum the travel time anomalies along each source-station ray path to generate realistic spatial correlations in the travel time anomalies. We sum the anomalies along the straight-line ray paths that would exist in the unperturbed half space. The three-dimensional (3-D) ray tracing would be more realistic, but our approximate approach is probably adequate to generate spatially correlated station terms of sufficient accuracy to address the relative location problem that is the focus of this paper. Figure 4 is an example of the isotropic 3-D velocity model, and

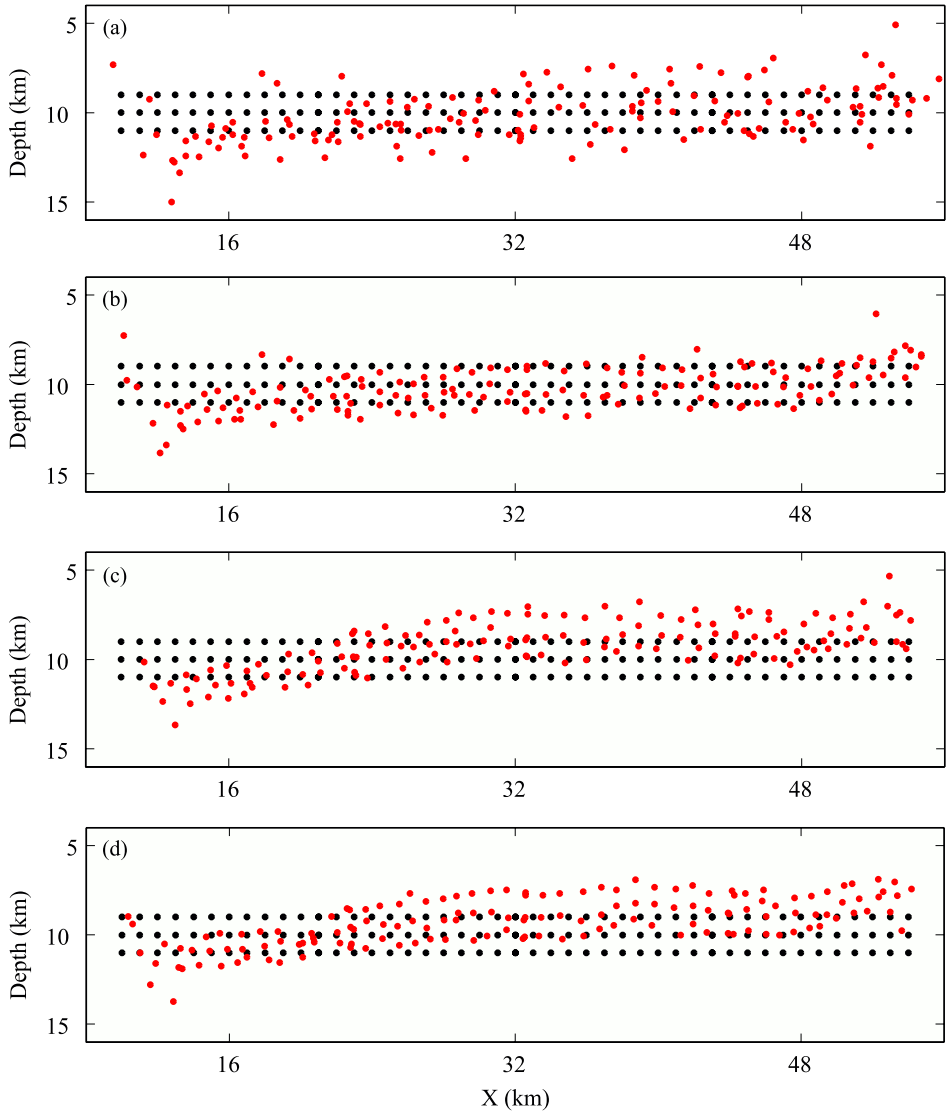

Figure 8. Cross section comparison of different location methods for one random realization of the three-dimensional velocity structure shown in Figure 4. Black dots are true locations, and red dots are computed locations. (a) Single event location. (b) DD location. (c) SSST location. (d) Shrinking box SSST location. In each case, the diagonal 147 events are shown, not the full 549.

Figure 5 shows the resulting station term field for a specific station location.

[46] As before, for each realization we generate 20 randomly located stations on the surface. We include 549 events in three layers of 183 events in the pattern shown in Figure 6. Events in each line of earthquakes are separated by 1 km. The three layers are located at depths of 9, 10, and 11 km. Picks for each event to each station are again generated with 0.67 probability for *P* and 0.5 probability for *S*. We apply a cutoff distance of 8 km for the DD and SSST methods. In most cases, this results in about 50 neighboring events in average for each target event in the calculations. For the shrinking box SSST method, we use a starting cutoff distance large enough to include all the events and then decrease the distance by a constant fractional change with each iteration to the final cutoff of 8 km.

[47] Figures 7 and 8 are the map views and cross sections of the relocation results of one random realization of the station locations and the 3-D velocity model. Notice the scatter in the single event locations compared to those of the other three methods. The difference between absolute and relative location errors is apparent in these plots. The reduced scatter in the relative locations sharpens the alignment of the events in each line, but the lines are still displaced from their true locations.

[48] Table 2 compares the location accuracy of the different methods, as measured by the RMS of absolute and relative location errors from 60 random realizations. The relative location errors are calculated for each event relative to a set of nearby events that are within a range of 2 km both in horizontal distance and in depth. As in the single event cluster test, the methods improve the absolute

Table 2. A Comparison of Four Location Techniques Applied to Synthetic Travel Time Data Predicted by Random 3-D Velocity Models[a]

	RMS of Absolute Error		RMS of Relative Error		
Method	Horizontal, km	Vertical, km	Horizontal, km	Vertical, km	Computation Time (Relative)
Single event	2.10	2.28	0.70	0.95	1
DD[b]	1.73	2.09	0.29	0.47	126
SSST	1.67	1.90	0.30	0.49	5
Shrinking box SSST	1.33	1.59	0.27	0.41	24

[a]We show absolute and relative location RMS errors from 60 random realizations. For a given event, the relative locations are with respect to the events within 2 km. Also we give the computation time of each technique relative to the single event location.
[b]No mean location shift during relocation.

location accuracy only slightly but significantly improve the relative location accuracy. There are greater errors in the vertical direction than in the horizontal direction. Finally, the shrinking box SSST method produces a slight, but noticeable improvement in absolute location accuracy compared to the other methods. We do not completely understand why the shrinking box method provides this advantage; it seems to respond to some of the longer-wavelength structure in the source-specific station term field that is not included in the DD or SSST methods when a fixed event separation distance is applied. We performed some experiments to see if methods that apply a distance weighting function to the SSST smoothing operator (i.e., weighting the nearby events more, the distant events less, with the weight a smooth function of distance rather than a simple cutoff at a fixed distance). However, we were not able to achieve results as good as the shrinking box method with this approach. We also performed some experiments in implementing the shrinking box approach to the DD method by reducing the range cutoff with increasing iteration number. We obtained some improvements in relative location accuracy, but the absolute locations were not as good as those given by the shrinking box SSST method.

[49] Table 2 also lists a measure of the computation time for the different methods, relative to single event location. In our implementation, the SSST technique is significantly faster than the DD method. This is because the SSST event locations at each iteration are still performed separately, so the run timescales approximately as the number of iterations (the station term calculation at each iteration is very fast compared to the location part of the calculation). In contrast, in the DD approach the location of every event is linked to the locations of all of the other events. This significantly increases the computation time even when the conjugate gradient method is used to solve the linear system. However, these results should be considered approximate because run times often depend upon specific details of program construction.

[50] Many networks do not produce as many S picks as we used in our synthetic experiments. To test the performance of the algorithms under these conditions, we also perform experiments using synthetic data that only contained P picks. Although the relocations are not as good as those obtained with both P and S picks, especially in depth, we still come to the same conclusion that DD and SSST produce quite similar relocation results and that the SSST shrinking box method is slightly better than the conventional SSST.

4. Discussion

[51] The advantage of our synthetic tests over real earthquake data sets is that the true event locations are known so that it is possible to directly measure the location errors. Although it is not possible to simulate every feature that may exist in real data, we have attempted to include the main factors that affect event locations, including random picking errors, incomplete and irregular pick distributions, station terms, and general three-dimensional velocity structure. We have focused on methods (JS, DD, and SSST) that attempt to improve the relative location accuracy among nearby events by taking advantage of the correlated travel anomalies from these events to each station. Despite differences in how the methods work, they are all solving the same underlying problem. Indeed, it is possible to demonstrate that station term algorithms provide a least squares iterative solution to the same equations that are used in the JS and DD methods. Thus it is reassuring that all of the methods achieve comparable results when applied to identical synthetic data sets.

[52] The DD and SSST methods can be used for large numbers of distributed earthquakes. An advantage of the DD algorithm is that a documented program has been released to the seismology community [*Waldhauser*, 2001]. This code also can incorporate differential time measurements provided by waveform cross correlation, a feature that is not included in our study, which uses only arrival time picks. In principle, our synthetic tests could be adopted to include waveform cross-correlation data by including differential times of greater accuracy than the individual picks. This will be a goal of our future work, as well as experimenting with the effects of some of the adjustable parameters in the DD and SSST algorithms.

Appendix A: Convergence of Static Station Term Method

[53] Here we demonstrate the convergence of the static station term method. Equation (5) gives the linearized location problem for a set of $p = 1, 2, \cdots, P$ earthquakes recorded by K_T stations:

$$\mathbf{A}\,\Delta\mathbf{X} + \mathbf{B}\mathbf{s} = \Delta\mathbf{T}$$

where $\mathbf{\Delta T}$ is a N_T vector containing the arrival time residuals, $\mathbf{A}$ defines a $N_T \times M_T$ matrix containing the partial derivatives which are calculated at a set of initial location estimates, $\mathbf{\Delta X}$ is a M_T vector containing the changes in hypocentral parameters we wish to determine ($M_T = 4\times P$), and $\mathbf{s}$ is a K_T vector containing static station terms, each of which is a constant for all events recorded by a given station. $\mathbf{B}$ is a $N_T \times K_T$ matrix that selects the correct station term for each arrival time, i.e.,

$$\left[\mathbf{B_p}\right]_{ij} = \begin{cases} 1 & \text{when } \left[\Delta t_p\right]_i \text{ is from station } j \\ 0 & \text{otherwise} \end{cases}$$

[54] This is a coupled set of equations for the unknowns $\mathbf{\Delta X}$ and $\mathbf{s}$, which may be solved iteratively by first solving for the locations $\mathbf{\Delta X}$ while holding the station terms $\mathbf{s}$ fixed, then solving for the station terms $\mathbf{s}$ while holding the locations $\mathbf{\Delta X}$ fixed, etc. This method has been applied for many years in seismology and was described by *Frohlich* [1979]. Here we show that this method is equivalent to a power series, which should converge to the least squares solution for $\mathbf{\Delta X}$ and $\mathbf{s}$.

[55] For the first iteration, we begin with zero station terms:

$$\mathbf{s}_1 = \mathbf{0}$$

then we solve the single event location problem starting with the initial guess of the hypocenter parameters and data,

$$\begin{aligned} \mathbf{A}\,\mathbf{\Delta X}_1 &= \mathbf{\Delta T} \\ \mathbf{\Delta X}_1 &= \mathbf{A}^\dagger \mathbf{\Delta T} \end{aligned}$$

but since the data cannot be fit perfectly even with the adjusted locations, we will obtain a new set of travel time residuals $\mathbf{R}_1$. The next step is to solve for the station terms from the new locations.

$$\begin{aligned} \mathbf{R}_1 &= \mathbf{\Delta T} - \mathbf{A}\mathbf{\Delta X}_1 \\ &= \mathbf{\Delta T} - \mathbf{A}\mathbf{A}^\dagger \mathbf{\Delta T} \\ &= \left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\mathbf{\Delta T} \\ \mathbf{s}_2 &= \mathbf{B}^\dagger \mathbf{R}_1 \\ &= \mathbf{B}^\dagger\left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\mathbf{\Delta T} \\ \mathbf{\Delta T}_2 &= \mathbf{\Delta T} - \mathbf{B}\mathbf{s}_2 \\ &= \mathbf{\Delta T} - \mathbf{B}\mathbf{B}^\dagger\left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\mathbf{\Delta T} \\ &= \left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]\mathbf{\Delta T} \end{aligned}$$

where $\mathbf{R}_1$ is a travel time residual vector and dagger means generalized inverse.

[56] We continue with the second iteration as follows:

$$\begin{aligned} \mathbf{A}\,\mathbf{\Delta X}_2 &= \mathbf{\Delta T}_2 \\ \mathbf{\Delta X}_2 &= \mathbf{A}^\dagger \mathbf{\Delta T}_2 \\ &= \mathbf{A}^\dagger\left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]\mathbf{\Delta T} \\ \mathbf{R}_2 &= \mathbf{\Delta T}_2 - \mathbf{A}\mathbf{\Delta X}_2 \\ &= \mathbf{\Delta T}_2 - \mathbf{A}\mathbf{A}^\dagger \mathbf{\Delta T}_2 \\ &= \left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\mathbf{\Delta T}_2 \\ &= \left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]\mathbf{\Delta T} \\ \mathbf{s}_3 &= \mathbf{B}^\dagger \mathbf{R}_2 \\ &= \mathbf{B}^\dagger\left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\mathbf{\Delta T}_2 \\ &= \mathbf{B}^\dagger\left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]\mathbf{\Delta T} \\ \mathbf{\Delta T}_3 &= \mathbf{\Delta T}_2 - \mathbf{B}\mathbf{s}_3 \\ &= \mathbf{\Delta T}_2 - \mathbf{B}\mathbf{B}^\dagger\left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right)\mathbf{\Delta T}_2 \\ &= \left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]\mathbf{\Delta T}_2 \\ &= \left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]^2\mathbf{\Delta T} \end{aligned}$$

[57] This can be generalized to the $k + 1$ iteration ($k \geq 0$):

$$\begin{aligned} \mathbf{\Delta X}_{k+1} &= \mathbf{A}^\dagger \mathbf{\Delta T}_{k+1} \\ &= \mathbf{A}^\dagger\left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]^k\mathbf{\Delta T} \\ \mathbf{s}_{k+2} &= \mathbf{B}^\dagger \mathbf{R}_{k+1} \\ &= \mathbf{B}^\dagger\left(\mathbf{I}_{\mathbf{N_T}} - \mathbf{A}\mathbf{A}^\dagger\right) \\ &\quad \cdot \left[\mathbf{I}_{\mathbf{N_T}} - \mathbf{B}\mathbf{B}^\dagger + \left(\mathbf{B}\mathbf{B}^\dagger\right)\left(\mathbf{A}\mathbf{A}^\dagger\right)\right]^k\mathbf{\Delta T} \end{aligned}$$

[58] So far, we have shown the changes in hypocentral parameters $\mathbf{\Delta X}_k$ and station terms $\mathbf{s}_k$ can be expressed as vector sequences. To show that these sequences are convergent, we use the equivalence between least squares fitting and orthogonal projection. Our problem can be defined as follows: we have a data space $\mathbf{T}$ and two model spaces, $\mathbf{M_U}$ and $\mathbf{M_V}$, the space of the location vectors and the space of the station term vectors. The linear function $\Phi_\mathbf{U}$: $\mathbf{M_U} \rightarrow \mathbf{T}$ gives the data vector produced by each location vector, and $\Phi_\mathbf{V}$: $\mathbf{M_V} \rightarrow \mathbf{T}$ gives the data vector produced by each station terms vector. Let $\mathbf{U} = \Phi_\mathbf{U}(\mathbf{M_U})$ and $\mathbf{V} = \Phi_\mathbf{V}(\mathbf{M_V})$.

[59] If $\mathbf{t}$ is the observed data vector (i.e., $\mathbf{\Delta T}$ in our previous notation), then our iteration scheme proceeds as follows. First, we set the initial station term contribution to the data vector to zero:

$$\begin{aligned}
\mathbf{v}_1 &= \mathbf{0} \\
\mathbf{u}_1 &= \mathbf{P_U}(\mathbf{t}-\mathbf{v}_1) = \mathbf{P_U}(\mathbf{t}) \\
\mathbf{v}_2 &= \mathbf{P_V}(\mathbf{t}-\mathbf{u}_1) = \mathbf{P_V}(\mathbf{t}) - \mathbf{P_V}\mathbf{P_U}(\mathbf{t}) \\
&\vdots \\
\mathbf{v}_{k+1} &= \mathbf{P_V}(\mathbf{t}-\mathbf{u}_k) \\
\mathbf{u}_{k+1} &= \mathbf{P_U}(\mathbf{t}-\mathbf{v}_{k+1})
\end{aligned}$$

where $\mathbf{P_U}$ is the orthogonal projector of $\mathbf{T}$ onto $\mathbf{U}$ and $\mathbf{P_V}$ is the orthogonal projector of $\mathbf{T}$ onto $\mathbf{V}$. Note that $\mathbf{P_U} = \mathbf{AA}^\dagger$ and $\mathbf{P_V} = \mathbf{BB}^\dagger$ in our previous notation. Then for $k \geq 1$,

$$\begin{aligned}
\mathbf{u}_{k+1} &= \mathbf{P_U}[\mathbf{t}-\mathbf{P_V}(\mathbf{t}-\mathbf{u}_k)] \\
&= \mathbf{P_U}(\mathbf{t}) - \mathbf{P_U}\mathbf{P_V}(\mathbf{t}) + \mathbf{P_U}\mathbf{P_V}(\mathbf{u}_k) \\
&= \breve{\mathbf{u}} + \mathbf{P_U}\mathbf{P_V}(\mathbf{u}_k) \\
&= \left[\mathbf{I} + \mathbf{P_U}\mathbf{P_V} + \cdots + (\mathbf{P_U}\mathbf{P_V})^{k-1}\right]\breve{\mathbf{u}} + \mathbf{P_U}\mathbf{P_V}(\mathbf{u}_1)
\end{aligned}$$

where $\breve{\mathbf{u}} = \mathbf{P_U}(\mathbf{t}) - \mathbf{P_U}\mathbf{P_V}(\mathbf{t})$

$$\begin{aligned}
\mathbf{v}_{k+1} &= \mathbf{P_V}[\mathbf{t}-\mathbf{P_U}(\mathbf{t}-\mathbf{v}_k)] \\
&= \mathbf{P_V}(\mathbf{t}) - \mathbf{P_V}\mathbf{P_U}(\mathbf{t}) + \mathbf{P_V}\mathbf{P_U}(\mathbf{v}_k) \\
&= \mathbf{v}_2 + \mathbf{P_V}\mathbf{P_U}(\mathbf{v}_k) \\
&= \left[\mathbf{I} + \mathbf{P_V}\mathbf{P_U} + \cdots + (\mathbf{P_V}\mathbf{P_U})^{k-1}\right]\mathbf{v}_2
\end{aligned}$$

[60] They will converge if $\|\mathbf{P_V}\mathbf{P_U}\| < \mathbf{1}$ and $\|\mathbf{P_U}\mathbf{P_V}\| < \mathbf{1}$. In finite-dimensional spaces, these inequalities hold whenever $\mathbf{U} \cap \mathbf{V} = \{\mathbf{0}\}$. However, in our case there is a nonuniqueness problem in that part of the data vector produced by the model can come from either the location vector or the station term vector. For example, a constant time added to the data vector could correspond to a change in the earthquake origin times or a change in the station terms. For this reason, in general $\mathbf{U} \cap \mathbf{V} \neq \{\mathbf{0}\}$.

[61] In this case, let $\mathbf{W} = \mathbf{U} \cap \mathbf{V}$, $\mathbf{U} = \widetilde{\mathbf{U}} + \mathbf{W}$, and $\mathbf{V} = \widetilde{\mathbf{V}} + \mathbf{W}$, where $\widetilde{\mathbf{U}} \perp \mathbf{W}$ and $\widetilde{\mathbf{V}} \perp \mathbf{W}$. We then have $\widetilde{\mathbf{U}} \cap \widetilde{\mathbf{V}} = \{\mathbf{0}\}$. We can write $\mathbf{t} = \tilde{\mathbf{u}} + \tilde{\mathbf{v}} + \mathbf{w} + \tilde{\mathbf{t}}$, with $\tilde{\mathbf{u}} \in \widetilde{\mathbf{U}}$, $\tilde{\mathbf{v}} \in \widetilde{\mathbf{V}}$, $\mathbf{w} \in \mathbf{W}$, and $\tilde{\mathbf{t}} \in (\widetilde{\mathbf{U}} + \widetilde{\mathbf{V}} + \mathbf{W})^\perp$.

[62] Some remarks are

$$\begin{aligned}
\widetilde{\mathbf{U}} \cap \mathbf{V} &= \mathbf{U} \cap \widetilde{\mathbf{V}} = \{\mathbf{0}\} \\
\mathbf{P_U} &= \mathbf{P}_{\widetilde{\mathbf{U}}} + \mathbf{P_W} \\
\mathbf{P_V} &= \mathbf{P}_{\widetilde{\mathbf{V}}} + \mathbf{P_W} \\
\mathbf{P}_{\widetilde{\mathbf{U}}}\mathbf{P_W} &= \mathbf{P_W}\mathbf{P}_{\widetilde{\mathbf{U}}} = \mathbf{0} \\
\mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P_W} &= \mathbf{P_W}\mathbf{P}_{\widetilde{\mathbf{V}}} = \mathbf{0} \\
\mathbf{P_U}\mathbf{P_V} &= \mathbf{P}_{\widetilde{\mathbf{U}}}\mathbf{P}_{\widetilde{\mathbf{V}}} + \mathbf{P_W} \\
\mathbf{P_V}\mathbf{P_U} &= \mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}} + \mathbf{P_W}
\end{aligned}$$

[63] For the station terms vector we have

$$\begin{aligned}
\mathbf{v}_2 &= \mathbf{P_V}(\mathbf{t}) - \mathbf{P_V}\mathbf{P_U}(\mathbf{t}) \\
&= [\mathbf{P}_{\widetilde{\mathbf{V}}}(\mathbf{t}) + \mathbf{P_W}(\mathbf{t})] - [\mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}}(\mathbf{t}) + \mathbf{P_W}(\mathbf{t})] \\
&= \mathbf{P}_{\widetilde{\mathbf{V}}}(\mathbf{t}) - \mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}}(\mathbf{t})
\end{aligned}$$

and for $k \geq 1$

$$\begin{aligned}
\mathbf{v}_{k+1} &= \mathbf{v}_2 + \mathbf{P_V}\mathbf{P_U}(\mathbf{v}_k) \\
&= \mathbf{v}_2 + \mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}}(\mathbf{v}_k) + \mathbf{P_W}(\mathbf{v}_k)
\end{aligned}$$

Clearly $\mathbf{v}_2 \in \widetilde{\mathbf{V}}$, and if $\mathbf{v}_k \in \widetilde{\mathbf{V}}$ ($k \geq 2$), then $\mathbf{P_W}(\mathbf{v}_k) = 0$, so $\mathbf{v}_{k+1} \in \widetilde{\mathbf{V}}$. Thus, by induction, $\mathbf{v}_k \in \widetilde{\mathbf{V}}$ for all k, $\mathbf{P_W}(\mathbf{v}_k) = 0$, and for $k \geq 1$

$$\begin{aligned}
\mathbf{v}_{k+1} &= \mathbf{v}_2 + \mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}}(\mathbf{v}_k) + \mathbf{P_W}(\mathbf{v}_k) \\
&= \mathbf{v}_2 + \mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}}(\mathbf{v}_k) \\
&= \left[\mathbf{I}_{\widetilde{\mathbf{V}}} + \mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}} + \cdots + (\mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}})^{k-1}\right](\mathbf{v}_2)
\end{aligned}$$

Since $\widetilde{\mathbf{U}} \cap \widetilde{\mathbf{V}} = \{\mathbf{0}\}$, $\|\mathbf{P}_{\widetilde{\mathbf{V}}}\mathbf{P}_{\widetilde{\mathbf{U}}}\| < 1$, so this series converges.

[64] For the location vector, we have

$$\mathbf{u}_{k+1} = \mathbf{P_U}(\mathbf{t}-\mathbf{v}_{k+1})$$

Since $\{\mathbf{v}_1, \mathbf{v}_2, \ldots\}$ converges and $\mathbf{P_U}$ is continuous, then $\{\mathbf{u}_1, \mathbf{u}_2, \ldots\}$ also converges.

[65] Our proof for convergence is limited to the least squares (L2 norm) solution of the static station term case described in the text. It can be shown that the proof fails for all non-Euclidian norms, but we do not know whether the result itself fails. We have found in practice (for both real and synthetic data) that convergence is quite rapid, typically being achieved in five iterations or less. We have also noticed that this iterative method seems to work for L1 norm locations algorithms as well, although we have not found a formal proof of convergence. Finally, this approach forms the basis for the SSST algorithm, which, although more complicated than static station terms, also achieves convergence in a small number of iterations in our tests on

synthetic data. The SSST algorithm applied by *Richards-Dinger and Shearer* [2000] to the southern California seismic catalog used a grid-search L1 norm approach and achieved a reasonably stable result after 10 iterations.

[66] **Acknowledgments.** We thank George Backus and Bob Parker for advice regarding the convergence properties of the iterative solution discussed in Appendix A. Cecily Wolfe and an anonymous reviewer provided constructive comments. Funding for this research was provided by NEHRP/USGS grant 03HQPA0001. This research was also supported by the Southern California Earthquake Center, which is funded by NSF Cooperative Agreement EAR-0106924 and USGS Cooperative Agreement 02HQAG0008. This is SCEC contribution 782.

References

Douglas, A. (1967), Joint epicentre determination, *Nature*, *215*, 47–48.

Evernden, J. F. (1969), Precision of epicentres obtained by small numbers of world-wide stations, *Bull. Seismol. Soc. Am.*, *59*, 1365–1398.

Freedman, H. W. (1967), A statistical discussion of *P* residuals from explosions, Part II, *Bull. Seismol. Soc. Am.*, *57*, 545–561.

Frohlich, C. (1979), An efficient method for joint hypocenter determination for large groups of earthquakes, *Comput. Geosci.*, *5*, 387–389.

Jordan, T. H., and K. A. Sverdrup (1981), Teleseismic location techniques and their application to earthquake clusters in the south-central Pacific, *Bull. Seismol. Soc. Am.*, *71*, 1105–1130.

Lilwall, R. C., and A. Douglas (1970), Estimation of *P*-wave travel times using the joint epicentre method, *Geophys. J. R. Astron. Soc.*, *19*, 165–181.

Pavlis, G. L., and J. R. Booker (1983), Progressive multiple event location (PMEL), *Bull. Seismol. Soc. Am.*, *73*, 1753–1777.

Pavlis, G. L., and N. B. Hokanson (1985), Separated earthquake location, *J. Geophys. Res.*, *90*, 12,777–12,789.

Pujol, J. (1988), Comments of the joint determination of hypocenters and station corrections, *Bull. Seismol. Soc. Am.*, *78*, 1179–1189.

Richards-Dinger, K., and P. Shearer (2000), Earthquake locations in southern California obtained using source-specific station terms, *J. Geophys. Res.*, *105*, 10,939–10,960.

Rubin, A. M., D. Gillard, and J.-L. Got (1999), Streaks of microearthquakes along creeping faults, *Nature*, *400*, 635–641.

Shearer, P. M. (2002), Parallel fault strands at 9-km depth resolved on the Imperial Fault, southern California, *Geophys. Res. Lett.*, *29*(14), 1674, doi:10.1029/2002GL015302.

Smith, E. G. C. (1982), An efficient algorithm for routine joint hypocentre determination, *Phys. Earth Planet. Inter.*, *30*, 135–144.

Viret, M., G. A. Bollinger, J. A. Snoke, and J. W. Dewey (1984), Joint hypocenter relocation studies with sparse data sets—A case history: Virginia earthquakes, *Bull. Seismol. Soc. Am.*, *74*, 2297–2311.

Waldhauser, F. (2001), hypoDD: A program to compute double-difference hypocenter locations (hypoDD version 1.0, 3/2001), *U.S. Geol. Surv. Open File Rep.*, *01-113*.

Waldhauser, F., and W. L. Ellsworth (2000), A double-difference earthquake location algorithm: method and application to the northern Hayward Fault, CA, *Bull. Seismol. Soc. Am.*, *90*, 1353–1368.

Waldhauser, F., and W. L. Ellsworth (2002), Fault structure and mechanics of the Hayward Fault, California, from double-difference earthquake locations, *J. Geophys. Res.*, *107*(B3), 2054, doi:10.1029/2000JB000084.

Waldhauser, F., W. L. Ellsworth, and A. Cole (1999), Slip-parallel seismic lineations along the northern Hayward fault, California, *Geophys. Res. Lett.*, *26*, 3525–3528.

Wolfe, C. (2002), On the mathematics of using difference operators to relocate earthquakes, *Bull. Seismol. Soc. Am.*, *92*, 2879–2892.

G. Lin and P. M. Shearer, Institute of Geophysics and Planetary Physics, University of California San Diego, La Jolla, CA 92093, USA. (gulin@ucsd.edu; pshearer@ucsd.edu)