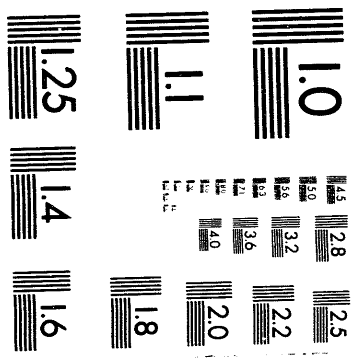


12/3-14-94 JS ①

UCRL-ID-116110

Location Capability of a Sparse Regional Network (RSTN) using a Multi-phase Earthquake Location Algorithm (REGLOC)

Lawrence Hutchings

January 1994

Lawrence Livermore National Laboratory

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the Laboratory.
Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This report has been reproduced
directly from the best available copy.

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information
P.O. Box 62, Oak Ridge, TN 37831
Prices available from (615) 576-8401, FTS 626-8401

Available to the public from the
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd.,
Springfield, VA 22161

Location Capability of a Sparse Regional Network (RSTN) using a Multi-phase Earthquake Location Algorithm (REGLOC)

Abstract

The Regional Seismic Test Network (RSTN) was deployed by the U.S. Department of Energy (DOE) to determine whether data recorded by a regional network could be used to detect and accurately locate seismic events that might be clandestine nuclear tests. The purpose of this paper is to evaluate the location capability of the RSTN. A major part of this project was the development of the location algorithm REGLOC and application of Basian *a prior* statistics for determining the accuracy of the location estimates. REGLOC utilizes all identifiable phases, including backazimuth, in the location. Ninty-four events, distributed throughout the network area, detected by both the RSTN and located by local networks were used in the study.

The location capability of the RSTN was evaluated by estimating the location accuracy, error ellipse accuracy, and the percentage of events that could be located, as a function of magnitude. The location accuracy was verified by comparing the RTSN results for the 94 events with published locations based on data from the local networks. The error ellipse accuracy was evaluated by determining whether the error ellipse (at the 95 and 99% confidence levels) includes the actual location. The percentage of events located was assessed by combining detection capability with location capability to determine the percentage of events that could be located within the study area. It is estimated that 100% of events above magnitude 3.0 could be detected and located with the RSTN.

Events were located with both an average crustal model for the entire region, and with regional velocity models along with station corrections obtained from master events. Most events with a magnitude <3.0 can only be located with arrivals from one station. Their average location errors are 453 and 414 km for the average- and regional-velocity model locations, respectively. Single station locations are very unreliable because they depend on accurate backazimuth estimates, and backazimuth proved to be a very unreliable computation. This is partially due to the fact that only small events required a single station location and that these also had a very poor SNR. However, even events with high SNRs have a backazimuth error of ±15°. Considering that the location error linearly increases with distance when using backazimuth estimates, this too suggests that backazimuth is a very poor measurement to use.

Fifty percent of the events with magnitudes of 3.0–3.5 are located with two stations. The average location errors when locating with two stations are 85 and 68 km for the average- and regional-velocity model locations, respectively. Locations with two stations are almost as accurate as locations with three or more stations; this is because backazimuth is not primary to the location computation, but only acts to distinguish between ambiguous interpretations. Ninety percent of the events above a magnitude of 3.5 are located with three or more stations. Their average location errors are 73 and 35 km, for the average- and regional-velocity model locations, respectively. The average location accuracy for events above a magnitude of 3.5 does not depend on magnitude.

Fifty-six percent of the events located with the average crustal model and 82% of the events located with regional velocity models have locations that fall within the 95% confidence error ellipses. Error ellipse estimates only account for the random errors in location. This suggests that including regional velocity models and calibrated travel times from master events does not account for all of the systematic errors. Most error ellipses that do not include the actual location of events are close to the event. Seventy-five and 86% of events fall within the 99% confidence error ellipses for the average and regional crustal models, respectively. For purposes of test ban treaty verification it is concluded that 100% of events above magnitude 3.0 could be detected and located with accuracy less than 68 km, and 99% error ellipses calculated are reliable at the 86% confidence level. Events below magnitude 3.0 cannot be fully detected and their location accuracy is fairly poor.

MASTER

Introduction

Seismic data can be used to monitor and verify treaties limiting or banning nuclear tests. The Regional Seismic Test Network (RSTN) was deployed by the U.S. Department of Energy (DOE) to determine whether data recorded by a regional network could be used to detect and accurately locate seismic events that might be clandestine nuclear tests. The RSTN was established as part of the U.S. Department of Energy's Test Ban Treaty Verification Program as a prototype of a network that could be installed in the former Soviet Union to monitor a Comprehensive Test Ban Treaty (CTBT) or Threshold Test Ban Treaty (TTBT). Monitoring for CTBT or TTBT will require evaluating approximately 50,000 seismic events with a magnitude $M \geq 2.5$ each year; one technique that could drastically reduce this computational effort is a discrimination method based on location. The RSTN was configured to detect possible clandestine nuclear tests and to provide data for preliminary discrimination based on location. However, to reliably monitor these treaties, the detection and location capability of the monitoring network must be well defined; the purpose of this study was to evaluate the usefulness (or location capability) of RSTN data by estimating the location accuracy, error ellipse accuracy, and the percentage of events located as a function of magnitude. The techniques developed for this study can also be applied to more conventional earthquake studies and may be useful in containing the proliferation of nuclear weapons.

The location accuracy is verified by comparing the RTSN results for 94 events to published locations based on data from local networks. These 94 events were distributed throughout the RSTN area. The error ellipse accuracy is evaluated by determining whether the error ellipse at the 95 and 99% confidence levels includes the actual location. The error ellipse reliability is more important to test ban treaty verification than the location estimate. Even when the location is not accurate, if it can be reliability assumed that an event occurred within an error ellipse, then discrimination based on location can be a very powerful technique. The percentage of events that can be located is assessed by combining detection capability with location capability to determine the percentage of events that could be located within the study area.

The RSTN consists of five broadband, three-component, seismic stations located in the United States and Canada: Tennessee (RSCP), Northwest Territories (RSNT), New York (RSNY), Ontario (RSON), and South Dakota (RSSD) (Figure 1; Table 1). Stations are distributed across North America in an area and geology roughly similar to that of the former Soviet Union. The RSTN system records signals from seismometers located in 100-m boreholes. After signal processing (Nakanishi et al., 1983), the instrument response is removed to obtain a broadband velocity that is flat between 0.5 and 16 Hz. A 40-sps digitized record of each event is saved for analysis.

For this study, the location capability is initially evaluated for an average earth structure for North America without station corrections—a situation that might exist during the early stages of monitoring. Then, the location capability is evaluated for specific regional velocity models with station corrections obtained from master events—a situation that would be possible after a period of operation. For very sparse data, Baysian *a priori* statistics are implemented to estimate travel-time variance and confidence-error ellipses. Further, statistics are developed for each individual phase at each station, and for each velocity model used. From these results, the most accurate error ellipse possible is estimated.

Because of the sparse station distribution, most events with magnitudes <3.0 are located with arrivals from one or two stations. For example, the only information available for a location may be the P_g and S_n phases and the backazimuth from one station 500 km distant, and the L_g phase and backazimuth from an additional station 2000 km distant. The algorithm (REGLOC) was developed to address a scenario in which earthquakes must be located with very sparse data. REGLOC utilizes multiple phases and backazimuth in an iterative least-squares location inversion. This, with signal processing to maximize the number of phases and backazimuths obtained, minimizes the magnitude threshold of location and maximizes the accuracy of location. At regional distances, identifiable phases include P_n, S_n, P_g, and L_g. Additional phases can be easily added to the location algorithm if necessary. Utilizing multiple phases also reduces inaccuracies that result from random errors. The L_g phase is a trapped higher mode Love-Rayleigh wave combination that arrives at the S_g wave velocity.

Location Algorithm

The generalized linear least-squares inversion location technique (Geiger, 1910; Lee and Lahr, 1971) is modified to include mixed-mode observational data (Jackson, 1972; Foy, 1975). Here, mixed-mode indicates arrival times and backazimuths. In addition, range estimates, as dependent on two arrival times, are automatically included. The basic mathematical principles are outlined below.

The observed i^{th} arrival time or backazimuth is τ_i, and the calculated value is t_i. The solution that minimizes the residuals

$$R_i = \tau_i - t_i(\phi_o, \lambda_o, z_o, t_o, \phi_s, \lambda_s) \tag{1}$$

is sought. Subscript s refers to station values, and ϕ_o, λ_o, z_o, and t_o are the true latitude, longitude, depth and origin time, respectively (henceforth referred to only as true location). The true location is expressed as a trial location (ϕ, λ, z, t) plus an error $(\delta\phi, \delta\lambda, \delta z, \delta t)$ called the adjustment vector:

$$\phi_o = \phi + \delta\phi$$

$$\lambda_o = \lambda + \delta\lambda$$

$$z_o = z + \delta z$$

$$t_o = t + \delta t.$$

The value t_i is expanded in a Taylor series about the trial location.

$$t_i = t_i' + \frac{\partial t_i}{\partial \phi}\delta\phi + \frac{\partial t_i}{\partial \lambda}\delta\lambda + \frac{\partial t_i}{\partial z}\delta z + \frac{\partial t_i}{\partial t}\delta t + \varepsilon_i', \tag{2}$$

where t_i' is the calculated travel time from the trial location, and ε_i' are higher order terms. The error is expressed as:

$$\varepsilon_i = \tau_i - t_i' + \varepsilon_i' = R_i' - \frac{\partial t_i}{\partial \phi}\delta\phi - \frac{\partial t_i}{\partial \lambda}\delta\lambda - \frac{\partial t_i}{\partial z}\delta z - \frac{\partial t_i}{\partial t}\delta t, \tag{3}$$

where R' is the residual computed for the trial location. Equation (3) is expressed in matrix form:

$$Gm = d, \tag{4}$$

where G is the operator matrix acting on the unknown column model matrix $m = [\delta t, \delta\phi, \delta\lambda, \delta z]^T$, d is the data matrix of travel time residuals. An estimate of m is obtained by:

$$m = [G + \beta I]^{-1} d, \tag{5}$$

where I is the identity matrix to dampen the solution (Menke, 1984, sec 4.7.3), and β is the damping factor. The necessary adjustment vector is solved by requiring

$$\sum_i w_i \varepsilon_i^2 = \text{a minimum}, \tag{6}$$

where w_i is the weight, and the sum is over i observations. Applying Eq. (6) to Eq. (3) leads to four equations and the four unknowns of the adjustment vector:

$$\frac{\partial}{\partial\phi}\sum\frac{w_i\varepsilon_i^2}{\sigma_i^p} = -\sum\frac{w_i a_i R_i'}{\sigma_i^p} + \sum\frac{w_i a_i}{\sigma_i^p}\delta t + \sum\frac{w_i a_i^2}{\sigma_i^p}\delta\phi + \sum\frac{w_i a_i b_i}{\sigma_i^p}\delta\lambda + \sum\frac{w_i a_i c_i}{\sigma_i^p}\delta z = 0$$

$$\frac{\partial}{\partial\lambda}\sum\frac{w_i\varepsilon_i^2}{\sigma_i^p} = -\sum\frac{w_i b_i R_i'}{\sigma_i^p} + \sum\frac{w_i b_i}{\sigma_i^p}\delta t + \sum\frac{w_i a_i b_i}{\sigma_i^p}\delta\phi + \sum\frac{w_i b_i^2}{\sigma_i^p}\delta\lambda + \sum\frac{w_i b_i c_i}{\sigma_i^p}\delta z = 0$$

$$\frac{\partial}{\partial z}\sum\frac{w_i\varepsilon_i^2}{\sigma_i^p} = -\sum\frac{w_i c_i R_i'}{\sigma_i^p} + \sum\frac{w_i c_i}{\sigma_i^p}\delta t + \sum\frac{w_i a_i c_i}{\sigma_i^p}\delta\phi + \sum\frac{w_i b_i c_i}{\sigma_i^p}\delta\lambda + \sum\frac{w_i c_i^2}{\sigma_i^p}\delta z = 0$$

$$\frac{\partial}{\partial t}\sum\frac{w_i\varepsilon_i^2}{\sigma_i^p} = -\sum\frac{w_i R_i'}{\sigma_i^p} + \sum\frac{w_i N}{\sigma_i^p}\delta t + \sum\frac{w_i a_i}{\sigma_i^p}\delta\phi + \sum\frac{w_i b_i}{\sigma_i^p}\delta\lambda + \sum\frac{w_i c_i}{\sigma_i^p}\delta z = 0. \tag{7}$$

The constants:

$$a_i = \frac{\partial t_i}{\partial\phi}; b_i = \frac{\partial t_i}{\partial\lambda}; c_i = \frac{\partial t_i}{\partial z}, \tag{8}$$

N is the number of observations, and σ_i^p is the prior variance of the i^{th} observation and normalizes the system of equations for mixed-mode observations so that the residual for each data point is compared to its expected error (Jackson, 1972). Prior observations are used to estimate variance [outlined for Eq. (30) below] because with sparse data the variance estimate based on observations for any one earthquake are insufficient to obtain a good estimate of variance. Normalizing by variance is weighting the location estimate by the more reliable phases. This is different than the weights w_i applied, which are an estimate of the quality of a particular arrival.

If the trial location is too far from the actual location, the Taylor series is not adequate, and the solution is attempted again starting from the new location. This process is repeated until the adjustment vector reaches some minimum value.

Four observations are required for a solution; three are required if depth is held fixed. Multiple observations provide a redundancy that averages out random errors in arrival times. With weighted least squares, the sum of the number of phases times their weight must equal four or three, respectively.

Range estimates based on relative arrival times are automatically included in the inversion since all identifiable phases are used. Consider τ_i to be the relative arrival time of two phases, then

$$t_i'(\phi, \lambda, z, t, \phi_s, \lambda_s) = \Delta\left(\frac{1}{V_2} - \frac{1}{V_1}\right) + c_2 - c_1, \tag{9}$$

where c_1 and c_2 are delay times (zero for direct arrivals), and V is velocity. Similarly, range estimates based on arrival times at two stations are:

$$t_i'(\phi, \lambda, z, t, \phi_{s1}, \lambda_{s1}, \phi_{s2}, \lambda_{s2}) = \frac{\Delta_2}{V_2} - \frac{\Delta_2}{V_2} + c_2 - c_1, \tag{10}$$

both of which satisfy the linear combination of observed arrival times.

Mixed-mode solutions require expressing $t_i'(\phi,\lambda,z,t,\phi_s,\lambda_s)$ for the different observations and taking the partial derivatives to obtain the constants. However, this alone will improperly weight the observations. For example, backazimuth measurements will have a contribution to Eq. (3) expressed as:

$$R_i' = w_i(BAZ_{obs} - BAZ_{cal})\,. \tag{11}$$

However, consider the error in location from the backazimuth estimate to be the distance Δ_i from a point along the observed backazimuth direction; the same distance from the station to the trial location. Then, the error in terms of travel time is:

$$R_i' = w_i\frac{\Delta_i(BAZ_{obs} - BAZ_{cal})}{V_i}\,. \tag{12}$$

The small angle approximation for $\sin(BAZ_{obs} - BAZ_{cal})$ is applied. Then a residual based on backazimuths only is improperly weighted by V_i/Δ_i.

The partial derivatives and backazimuth are calculated for a geocentric earth model (Herrmann, 1982) with the relations for distance:

$$\cos\Delta = \cos\phi_e\cos\phi_s\cos(\lambda_e - \lambda_s) + \sin\phi_e\sin\phi_s \tag{13}$$

and relations for backazimuth are:

$$\sin BAZ = \frac{-\cos\phi_e\sin(\lambda_s - \lambda_e)}{\sin\Delta},$$

$$\cos BAZ = \frac{\sin\phi_e - \cos\Delta\sin\phi_s}{\sin\Delta\cos\phi_s}\,. \tag{14}$$

There are several conditions on the backazimuth relations if they are used separately, but

$$BAZ = \text{atan}\left(\frac{\sin BAZ}{\cos BAZ}\right) \tag{15}$$

has no restrictions if the sign of the numerator and denominator are preserved, as in the $\text{atan2}(\sin BAZ, \cos BAZ)$ FORTRAN function.

In the above relations, ϕ and λ are geocentric latitude and longitude, respectively, and Δ is distance in degrees (1° = 111.12 km). The subscript e refers to event location, and the subscript s refers to station location.

Geographic latitude and longitude (ϕ', λ') are defined with the convention:

$$-90° \le \phi' \le 90°, \text{ positive for north}, \tag{16}$$

and

$$-180° < \lambda' \le 180°, \text{ positive for east}.$$

Geocentric latitude and longitude are obtained from the relations:

$$\tan\phi = (1-\gamma)^2 \tan\phi' = 0.993277 \tan\phi',$$
$$\lambda = \lambda', \tag{17}$$

where γ is the correction factor for earth ellipticity ($\gamma = 0.003367003$).

The partial derivatives of travel times are

$$\frac{\partial t_i}{\partial \phi_e} = \frac{\sin\phi_e \cos\phi_s \cos(\lambda_e - \lambda_s) - \cos\phi_e \sin\phi_s}{V_k \sin\Delta} \tag{18}$$

and

$$\frac{\partial t_i}{\partial \lambda_e} = \frac{\cos\phi_e \cos\phi_s \sin(\lambda_e - \lambda_s)}{V_k \sin\Delta} \tag{19}$$

Units are s/deg, which are controlled by units of V_k. For backazimuth estimates,

$$\frac{\partial BAZ_i}{\partial \phi_e} = \frac{-\cos\phi_e}{\sin BAZ \cos\phi_s \sin\Delta} \tag{20}$$

and

$$\frac{\partial BAZ_i}{\partial \lambda_e} = \frac{\cos(\lambda_e - \lambda_s)\cos\phi_e}{\cos BAZ \sin\Delta}, \tag{21}$$

where BAZ_i distinguishes backazimuth observations previously identified only as t_i. These partials are unitless (deg/deg). Note that the relations:

$$\frac{\partial \cos\theta}{\partial \chi} = \sin\theta \frac{\partial\theta}{\partial\chi} ; \frac{\partial \sin\theta}{\partial \chi} = \cos\theta \frac{\partial\theta}{\partial\chi} \tag{22}$$

make the partial derivatives simple to compute.

The algebraic solution from one station where backazimuth and distance are known is:

$$\phi_e = \sin^{-1}[\sin\phi_s \cos\Delta + \cos\phi_s \sin\Delta \cos BAZ] \tag{23}$$

$$\lambda_e = \lambda_s \pm \cos^{-1}\left[\frac{\cos\Delta - \sin\phi_s \sin\phi_e}{\cos\phi_s \cos\phi_e}\right], \tag{24}$$

where the second term on the right is added if BAZ is ≤180.0°, and it is subtracted if BAZ is >180.0°.

Error Estimate

Standard means by which the location error is estimated for least-squares problems require some modification when mixed-mode and sparse data are used. The errors in location are assumed to be a result of random errors from misidentification of phase arrival times and heterogeneities in the velocity structure and the result of systematic errors from incorrectly estimating the velocity model. In the following explanation, the conventional method for estimating location errors resulting from random errors in location algorithms is outlined, and then the method for estimating location errors using sparse data is discussed. Systematic errors are addressed later. The variance is an indication of the random error and is generally obtained by:

$$\sigma_d^2 = \frac{1}{N-L}\sum_{i=1}^{N} w_i R_i^2, \tag{25}$$

for N observations and L model parameters. The subscript d refers to the estimate from the data used in the location only; σ_d^2 is the data variance. Subscripts are used here for d and p instead of superscripts for visual clarity. If the variance is computed from prior observations, as discussed below, then Eq. (25) represents σ_p^2.

The covariance matrix is generally used to estimate errors in model parameters. The covariance matrix is:

$$V = \sigma_d^2 [G^T G]^{-1}, \tag{26}$$

and the diagonals are variance estimates for model parameters.

The joint confidence ellipsoid for model parameters is defined by:

$$(m-\bar{m})^{-1} V^{-1} (m-\bar{m}) = c_L^2, \tag{27}$$

where $\bar{m}$ is the unknown true model parameter, and c_L depends on the confidence coefficient:

$$c_L^2 = L\sigma_d^2 F(p; L, N-L), \tag{28}$$

where $F(p;L,N-L)$ is the F statistic with L and $N-L$ degrees of freedom at the 100% confidence level, and p is the confidence probability sought (Flinn, 1965).

This approach is adequate when an event is located by a large number of arrivals so that σ_d^2 is a reasonable estimate of the true variance, but for sparse data this is not generally true. Jordan and Sverdrup (1981) have addressed this problem, and their approach is used here. They suggest adopting the Baysian technique of imposing a prior distribution based on previous recordings of phase travel times to estimate the true variance:

$$\sigma^2 = \frac{(K-L)\sigma_p^2 + (N-L)\sigma_d^2}{K+N-L} \tag{29}$$

for K and N prior and data observations, respectively.

In the situation studied here where multiple phases and backazimuths are used, each phase may have a different variance for each station and velocity model used. In REGLOC, prior variance is only computed from the phases and stations used in the location. The prior variance is computed for good arrivals only, and all weights are assigned 1.0, except when obtaining prior variance for backazimuth, as discussed below.

Although the prior error in backazimuth may be estimated, the effect on earthquake location is distance dependent. Therefore, the sum of residuals for a particular station from backazimuth is converted to seconds by:

$$R_i^2 = \sum_K \frac{\Delta_i^2}{V^2} w_i R_i'^2, \tag{30}$$

where Δ_i is the distance from the station to the event location. Backazimuth estimates are so unreliable that they are only used in locations with three or fewer stations. When locating with two or three stations, the backazimuth estimates are first weighted to 0.1, then distance weighted so that $w_i = (0.1 * 200\text{ km})/\Delta_i$. Backazimuth residuals are not very reliable for computing error ellipses either. For example, the root-mean-square contribution, without weighting, from an arrival 2000 km distant with a BAZ prior error of 20° would be 196 s. This leads to unrealistically large error ellipses. Therefore, errors to backazimuth are only used to contribute to a prior variance when locating with a single station, and then an *ad hoc* weighting of 0.1 is applied to bring error ellipses to a size that experimentally fit the 95% confidence requirement.

The G matrix was previously normalized for the mixed-mode problem by dividing the contribution from each observation by σ^p to obtain units of 1/deg. Now, the G matrix is multiplied by σ to arrive at s^2/deg units for computations of the error ellipse. The joint confidence location error ellipse is determined from E, the second order covariance matrix of horizontal location parameters (ϕ,λ), and the variance σ. Then, from Eqs. (27) and (28):

$$c_2^2 = l_1 \Delta M_\phi^2 + l_2 \Delta M_\lambda^2, \tag{31}$$

with

$$c_2^2 = 2F(0.95;2,K+N-2)\sigma$$

$$l_1 = E_{11}^{-1}\cos^2\alpha + 2E_{12}^{-1}\sin\alpha\cos\alpha + E_{22}^{-1}\sin^2\alpha$$

$$l_2 = E_{11}^{-1}\sin^2\alpha - 2E_{12}^{-1}\sin\alpha\cos\alpha + E_{22}^{-1}\cos^2\alpha. \tag{32}$$

F is obtained from tables (see CRC, Standard Math Tables, p. 272), where for example $F(0.95;2,15) = 3.68$. α is the angle from north for the Γ_1 axis:

$$\alpha = 0.5\tan^{-1}\left[\frac{2E_{12}^{-1}}{E_{11}^{-1}-E_{22}^{-1}}\right]. \tag{33}$$

The length of the semiaxes are:

$$\Gamma_1 = 111.12\left(\frac{c_2^2}{l_1}\right)^{1/2},$$

and

$$\Gamma_2 = 111.12\left(\frac{c_2^2}{l_2}\right)^{1/2}. \tag{34}$$

Γ_1, Γ_2 are multiplied by 111.12 km/deg to obtain the length of possible location error (in kilometers) for an epicenter at the center of the ellipse; these with α are output in REGLOC. REGLOC also provides the 95% limits of the model parameters, assuming a Gaussian distribution:

$$\delta m = \sqrt{-2\sigma^2 ln\,[P(m)\,(2\pi)^{1/2}\sigma]} \tag{35}$$

in units of degrees.

RSTN Location Capability

The data used for evaluating the location capability of seismic recordings made by the RSTN consist of records of 94 earthquakes identified by local network bulletins and recorded on the RSTN (Jarpe et al., 1986); epicenters are plotted in Figure 2 and are listed in Table 2. The events used in this analysis were randomly selected, but an even distribution in location and magnitude was attempted.

The spatial limits of events considered in this study are the western boundary of the Rocky Mountains, the East Coast, the northern extent of the Canadian land mass, and the southern extent of the United States. This generally encompasses the Canadian Shield and is approximately equal to the land mass of the former Soviet Union. Figure 2 shows that many of the events are located outside the perimeter of the network. In addition, there is an uneven distribution of stations over the region. The eastern portion of Canada has poor station coverage, and the central portion of the United States has the highest density of stations.

Phase detection and identification are done in a manner that optimizes the number of phases used. Time series files are bandpassed in four frequency bands (0.5–1, 1–2, 2–4, and 4–8 Hz) and then are plotted. The expected arrival time of the phases are marked on the records (based on the known location of the event), and the arrival times are picked by eye. Identifying the phases from prior knowledge is justified under the assumption that phase identification will be possible with signal processing and artificial intelligence programs (Jarpe et al., 1991). Location analysis is then performed with the best data that can be obtained. Phases are assigned a weight based on the signal-to-noise ratio as outlined in Table 3. Backazimuths are obtained from the MUSIC algorithm (Dowla and Harris, 1987). Appendix A lists the phase arrival times and backazimuths read for the 94 events.

All events with at least a minimum amount of information are located. Depths are constrained to 10 km, since depth cannot be resolved with arrivals at regional distances. Therefore, there are 3-degrees -of freedom in the locations. Table 4 lists the minimum information necessary to obtain an epicentral location for one, two, or three stations. Each column represents a combination of arrival times from any identifiable arrival (AT) and backazimuths (BAZ) that can be obtained at a single station. An X is placed on the contribution from a single station that is necessary for a location with the number of stations listed in column one. More than one backazimuth at one station is averaged to obtain a single value, and more than two arrival times at a single station are used to obtain an average distance estimate. Redundant information is not listed. For example, two arrival times and a backazimuth at one station are sufficient for a location, and this is not listed as a combination for two or three stations. At two stations, one arrival time and backazimuth at one station and a backazimuth at another are redundant to two backazimuths.

The location algorithm is used even if only the minimum number of phases or backazimuths is available, except for locating with one station, in which case an algebraic solution is obtained. Other locations are nonlinear in ϕ and λ through sines and cosines, and a solution with linear algebra is not possible.

Table 5 shows the number of events in each magnitude range, and the events that had at least the minimum information required for a location.

Location Accuracy with the Average Crustal Model

The average velocity model for each shield region used is listed in Table 6. Prior estimates of travel time variance are obtained by locating all of the events and by using the residuals obtained as input into Eq. (25). The location error estimates are then obtained by calculating the locations again. Table 7 shows the individual phase-travel-time residual squared and the number of readings for each phase at each station used for the error analysis. To avoid biasing the prior variance values, only residuals from locations using more than the minimum data required are used. Also, only residual values from phases that were used in the final solution are used. Residual squared values shown for backazimuth estimates are in degrees.

Table 8 is a summary of RSTN location information, the location error as compared to published reports, and to the maximum semi-axis of the error ellipse estimated by REGLOC. Ten of the ninety-four events had a location error greater than the maximum of the semi-axis of the error ellipse, and these ten event location errors were generally close to the predicted. The relation of the error ellipses to published locations is discussed later. Large location errors primarily occur when the locations were determined using information from one or two stations that rely on backazimuth estimates in the computations. Single station locations in general have the worst location error; the average location error for these events is 453 km. This is not surprising considering the nearly 20° un-reliability of backazimuth estimates. Two station locations are generally considerably better than single station locations since the backazimuth estimates are not primary to the location estimate but do help in distinguishing between dual solutions that result if only travel times are used. Locations based on two stations have an average location error of 85.0 km. The average location error for the events located with three or more stations is 73.4 km. For purposes of test ban treaty verification, single station locations have extreme limitations, but two station locations can reliably estimate the general location of an event.

Figure 3 is a plot of the location error as a function of magnitude. Most of the events with a magnitude <3.0 are located with one station and have the largest location error. Events with a magnitude 3.0 and greater have an average location error of 61 km (excluding the three outliers) and the location error is fairly independent of magnitude. Nineteen of the 38 events with magnitudes 3.0–3.6 are located with two stations, and 18 of the 20 events above magnitude 3.6 are located with three or more stations.

Figure 4 shows 95% confidence error ellipses for events using two stations to estimate their location and the actual location; Figure 5 shows error ellipses for events using three or more stations to estimate their location. Even though the absolute location errors are predicted by the error ellipses, some of the actual locations of the events are outside the error ellipses. This is a result of a systematic error in location resulting from an incorrect velocity model. Nine of the 26 events located with two stations and 20 of the 35 events located with three or more stations have locations outside the 95% confidence error ellipses. Figures 6 and 7 show 99% confidence error ellipses for the same events shown in Figures 4 and 5. Here, 5 and 9 events, respectively, fall outside the error ellipses. Examination of Figures 6 and 7 show that events that do not have locations within the error ellipses are very close to the error ellipses. Error ellipse accuracy is discussed in more detail later.

In summary, events with a magnitude <3.0 are principally located with one station and have extremely unreliable locations. Events with magnitude 3.0 and greater are principally located with two or more stations and have an average location accuracy of about 61 km. Ninety-five percent confidence limits are systematically displaced because of the inadequacy of the average crustal velocity model, and 99% error ellipses are a better estimate of the location for events using an average crustal model.

Location Accuracy with Regional Velocity Models and Station Corrections

Travel times from well-located large events and additional information about the regional geology can be used to further improve the location accuracy. Figure 8 shows the study area divided into four regions (two of the regions are further subdivided for station corrections) to allow a more detailed study of the location capability of the RSTN data. Events that have an initial location in one of the four regions are relocated with a velocity model for that region along with station corrections appropriate for events from that region. A master event located within each region, or subregion, is used to obtain travel time corrections to all stations. All events within the region are then relocated with the station's travel time corrections appropriate for the region.

Figure 1 shows the major crustal provinces for North America from Taylor and Qualheim (1983). Velocity models for the Churchill, Superior, Central, and Grenville provinces are listed in Table 9; they are described by the thickness of the crust above and below the Conrad discontinuity, the corresponding average velocity, and the velocity of the upper mantle. The Churchill province velocity model is obtained from the refraction studies of Prodehl (1970) and the Rayleigh wave dispersion of Keller et al. (1976). Both studies agree fairly well on the main features of the velocity structure. The structures for the Superior and Grenville provinces are obtained from the refraction studies of Berry and Fuchs (1973) and the receiver function studies of Owens et al. (1987). The structure for the Central province is obtained from Carts and Bollinger (1981). They obtain velocities of P_g, P_n, S_n, and L_g by plotting arrivals from a well-located M_L = 3.7 earthquake. They argue that the Conrad discontinuity is not well-defined or nonexistent in this region, and they develop a layer over the half- space velocity model. Even so, it is difficult to accept a continuous velocity for a 39 km increase in depth; therefore, an intermediate layer is included for this study.

Master events used to obtain travel time corrections for events are indicated by an asterisk in Table 2 and are plotted with a different symbol in Figure 8. Station corrections are obtained by locating events with the regional velocity model (without station corrections), and with *a priori* travel time residuals for the average earth model listed in Table 7. Table 10 lists the travel time residuals obtained from the master events that are used to obtain station corrections. No master event is used for the Churchill I area, because there were no well-recorded events. The location error between local network and REGLOC master event locations was small for all regions except for the Superior and Churchill III provinces master events, which had a location errors of 50 and 25 km, respectively. These events, and the resulting station corrections, were used anyway because in a situation where the actual earthquake location was not known, they would have been used.

To obtain Baysian statistics for the estimation of error ellipses, all events are first located with the appropriate velocity model and station corrections, but *a priori* travel time residuals for the average earth model listed in Table 7 are used to calculate σ_p to normalize the location equations [Eq. (25)]. Once all the events for a particular province or subprovince are located, then *a priori* travel time residuals particular for that region are calculated, these are listed in Table 11. Only phases with at least 3 residuals are used to calculate *a priori* values since the same residuals will be included again in the final runs and will effectively be used twice (although the final runs have different locations, and residuals, when the *a priori* values are used to calculate variance for inversion are changed). The master events are not used to calculate *a priori* values since their residuals are forced to be zero by the selection of the station corrections. Phases with zeros in Table 11 did not have enough phases recorded to get *a priori* values; those from the average model in Table 7 were used instead for location calculations. The events are then located again using the *a priori* statistics. Locations change slightly because of the different statistics used for normalization. The *a priori* residuals calculated for the Churchill II area are also used for the Churchill I province. Final locations and location error information are listed in Table 12.

Single station locations remain extremely unreliable, as they are when obtained with the average crustal model. The average location error for single station locations improved, but was 414 km. Locations with two stations improved, and the average location error was 68.3 km. For events located with three or more stations, and the average location error also improved and was 34.8 km. Events used as master events are included in computing average location errors since their locations were not fixed by the local network locations, and were determined in the same manner as other events. Location errors using regional velocity models and station corrections provides 20% and 53% improvement in location accuracy over the average velocity models for the 2 station and 3 or greater station locations, respectively. Figure 9 shows location error as a function of magnitude. For events with a magnitude of 3.0 and greater, the average location error is 55.3 and is fairly independent of magnitude.

Figures 10 and 11 show 95% confidence error ellipses for events using two stations and three or more stations, respectively, along with the actual location of the event. The error ellipses are actually a little larger than those obtained from the average crustal model; compare the last columns in Tables 8 and 12. The error ellipses are larger because the random errors have slightly increased with the use of the calibrated crustal models. However, the systematic errors have decreased. This is evident by the number of error ellipses that now include the actual locations of the events: 48 of the 59 events have error ellipses that include the actual locations of the events, as compared to 33 of the 59 events when the average crustal model is used. The fact that fewer than 95% of the events in either case do not fall within the 95% confidence ellipses is possibly because error ellipses only take into account random errors and the systematic errors have not all been accounted for. When only events located with three or more stations are used 28 of 33 events (85%) have locations within the 95% error ellipses (Figure 11), and indicate a reliability of error ellipse estimation for these events; only 16 of 33 the events fall within the 95% error ellipse for the average crustal model (Figure 5). Figures 12 and 13 show 99% confidence error ellipses for the same events shown in Figures 10 and 11, where 51 of the 59 events have error ellipses that include the actual locations of the events, as compared to 44 of the 59 events when the average crustal model is used. When only events located with three or more stations are used 30 of 33 events (90%) have locations within the 99% error ellipses (Figure 11), and indic 'e a reliability of error ellipse estimation for these events.

In summary, only events located with three or more stations, and using regional velocity models and master events have location error ellipse that are reliable, but only at the 85% and 90% level when 95% and 99% error ellipses are used, respectively. For events located with two stations only 77% of the events are located within the 95% confidence error ellipses, and statistics are not significantly improved by using 99% error ellipses. Similarly, for events located with the average crustal model only 65% and 48% of the events fall within the 95% confidence ellipses for the two and three or more station locations, respectively.

Percentage of Events Located

For purposes of test ban treaty verification, the percentage of events located refers to events that can occur anywhere within the network. This assumes an even distribution of events throughout the network. This is distinctive from percentage of actual events occurring within the network, which is biased by the distribution of seismic sources and differing seismicity rates.

Since the 94 events studied here are not evenly distributed in magnitude and location, a straightforward analysis will not yield a reliable assessment of the percentage of events that could be located. Rather, an analysis of detection capability along with location capability is used here for the assessment.

Jarpe et al. (1986) assessed the detection capabilities of RSTN data using the same data set as used for this study along with an additional 116 events. The time series were bandpassed and expected arrival times were marked on data files as described above. They then measured the maximum trace amplitudes on all three components near the time expected for P_n, P_g, S_n, and L_g. They calculated the signal-to-noise ratio (SNR) for each phase in each band on each component by dividing the maximum phase window amplitudes by the pre-event noise amplitude. A phase detection occurred when the SNR in the velocity window exceeded a threshold of 3.0. They found that 3.0 was the lowest value that provided real phase detections as opposed to spurious detections (based on a visual inspection of amplitude); they did not examine a false alarm rate. An SNR of 3:1 is a stiffer criteria than is necessary to obtain information that can be used for a location, but is a good minimum SNR for ensuring good data, a minimum number of spurious arrivals, and good quality backazimuths. This then is accepted as a good criteria for assessing location capability.

Figure 14 (from Jarpe et. al., 1986) shows the percentage of phases (of 848 possible) detected versus magnitude and recording distance. This is irrespective of source region and recording station. This, then represents an average value for the network and does not account for regional differences in attenuation. Considering that the longest midpoint distance between any two stations of the RSTN is about 1200 km (the northern extreme of eastern Canada is greater), the point along the curves in Figure 14 at 11° represents the minimum number of phases detected from events anywhere within the network. For example, a minimum of 40% of the phases for magnitude $M = 2.5–3.0$ and 80% of the phases for magnitude $M = 3.0–3.5$ should be recorded. Since we only need 2 phases to obtain a location (Table 4), 100% of the events greater than magnitude 3.0 and slightly less 100% of the total number of events greater than magnitude 2.5 should have been located. The results of this study indicate fewer than expected percentage of events were actually located; as is evident from Table 5, where 89% of the events with magnitude greater than 2.5 were actually located. Regional variations in recording capability and random or systematic interruptions in recording can also affect the results.

Figure 15 shows the Jarpe et al. (1986) analysis of percentage of detections versus recording distance, irrespective of magnitude, for each station. Considering a fairly even distribution of sources for each station, this indicates the variation of recording capability resulting from regional differences in geologic structure. Figure 1 shows the major geologic-seismic velocity provinces of the RSTN region (from Owens et al., 1987). Station RSCP and RSNY, for example, record arrivals traversing the Grenville Orogene belt, which has a lower Q factor than the interior U.S. and Canadian shields, and a significantly lower percentage of arrivals are recorded. The relative percentage of events located by each station versus distance is taken as the effect of the regional geologic structural differences. For example, the average percentage detected at 1000 km is 75%, and station RSNT detects 50%. It then has a 33% depletion in recording because of regional differences. Since the curves in Figure 15 are fairly parallel, this suggests that the effect is fairly uniform for all distances. To incorporate regional differences in Figure 14, the curves are multiplied by the depletion or amplification factor, from the average, to represent that from an individual station. Figure 14 modified for station RSTN, for example, would indicate that 34% of the phases for magnitude 2.5–3.0 and 50% of phases for magnitude 3.0-3.5 would be recorded at 1200 km. Considering these factors for all stations it is estimated that about 90% of the events with magnitude greater than or equal to 2.5 and 100% of events greater than or equal to 3.0 should have been located. Since, as discussed previously, fewer than 100% of the events greater than or equal to magnitude 2.5 (Table 5) were actually located, it is concluded that this is a result of spurious interruptions in data recording or anomalously attenuating travel paths. Figure 16 shows the estimation of the number of events that should be located as a function of distance. This is obtained by modifying table 14 for the differences in regional recording capability as described, and integrating over all recording distances. Figure 16 also shows the results from Table 5. The differences between the two curves is possibly due to spurious interruptions in recording.

Discussion and Conclusions

It is rather surprising that with such a sparse network of the five three-component stations covering most of North America that most events with magnitude greater than 2.5 can be located, and with a fairly reliable accuracy. It was found that approximately 90% of all magnitude ≥2.5 events occurring anywhere within the RSTN can be detected and located and that approximately 93% of events with a magnitude ≥3.0 can be detected and located. Further, tests of phase recordings as a function of magnitude and distance suggest that the RSTN should detect and locate 100% of the events with a magnitude $M \geq 3.0$. This discrepancy is attributed to random interference and interruptions in data collection and to the variability in the quality of the arrivals. Even so, some phases used in this study were generated by events with magnitudes <3.0 that were recorded over 3000 km from where they occurred, yet the data still provided enough information to locate the event within 50 km. Much of this capability is attributed to recording the L_g phases, which in a shield region channels the S-wave energy and has very low attenuation. All phases are generally recorded for events with magnitudes >3.5. Forty-two percent of all phases recorded are L_g phases; the S_n phase is the least useful with only 10% recorded for distances >200 km. The implications are obvious—this study would have significantly different results if we were examining a region that provided poor transmission of the L_g phase.

Most events with a magnitude <3.0 can only be located with arrivals from one station. Their average location errors are 453 and 414 km for the average- and regional-velocity model locations, respectively. Single station locations are very unreliable because they depend on accurate backazimuth estimates, and backazimuth proved to be a very unreliable computation. This is partially due to the fact that only small events required a single station location and that these also had a very poor SNR. However, even events with high SNRs have a backazimuth error of ±15°. Considering that the location error linearly increases with distance when locating with backazimuth estimates, this too suggests that backazimuth is a very poor measurement to use.

Fifty percent of the events with magnitudes of 3.0–3.5 are located with two stations. The average location errors when locating with two stations are 85 and 68 km for the average- and regional-velocity model locations, respectively. Locations with two stations are almost as accurate as locations with three or more stations; this is because backazimuth is not primary to the location computation, but only acts to distinguish between ambiguous interpretations.

Ninety percent of the events above a magnitude of 3.5 are located with three or more stations. Their average location errors are 73 and 35 km, for the average- and regional-velocity model locations, respectively. The average location accuracy for events above a magnitude of 3.5 does not depend on magnitude. Error ellipse estimates only account for the random errors in location. Fifty-six percent of the events located with the average crustal model and 82% of the events located with regional velocity models have locations that fall within the 95% confidence error ellipses. This suggests that including regional velocity models and calibrated travel times from master events does not account for all of the systematic errors. Most error ellipses that do not include the actual location of events are close to the event. Seventy-five and 86% of events fall within the 99% confidence error ellipses for the average and regional crustal models, respectively. For purposes of test ban treaty verification, then, it is recommended that 99% error ellipses be calculated, but only considered to be reliable at the 75 and 86% confidence level. For purposes of test ban treaty verification it is concluded that 100% of events above magnitude 3.0 could be detected and located with accuracy less than 68 km, and 99% error ellipses calculated are reliable at the 86% confidence level. Events below magnitude 3.0 cannot be fully detected and their location accuracy is fairly poor.

References

Bratt, S. R., and T. C. Bache (1988), "Locating Events with a Sparse Network of Regional Arrays," *Bull. Seismol. Soc. Am.* **78**, 780–798.

Berry, M. J., and K. Fuchs (1973), "Crustal Structure of the Superior and Grenville Provinces of the Northeastern Canadian Shield," *Bull. Seismol. Soc. Am.* **63**, 1939–1432.

Carts, D. A., and G. A. Bollinger (1981), "A Regional Crustal Model for the Southeastern United States," *Bull. Seismol. Soc. Am.* **63**, 1939–1432.

Dowla, F.U. and D.B. Harris (1987) Direction estimation of vector planewave fields, *Proc.* IEEE *International Conference on Acoustics Speech and Signal Processing* 2225-2228.

Flinn, E. A. (1965), "Confidence Regions and Error Determinations for Seismic Event Location," *Rev. Geophy.* **3**(1), 157–185.

Foy, W. H. (1976), "Position-Location Solutions by Taylor-Series Estimation," *IEEE Trans. on Aerosp. and Elect. Sys.* **AES-12**, 187–193.

Herrmann, R. B. (1982), "Digital Processing of Regional Network Data," *Bull. Seismol. Soc. Am.* **72**, part B, S261–S276.

Jarpe, Stephen P., Carolyn Baily, Brenda C. Bowman, Elizabeth K. Foote, Keith J. MacKay, Robert M. Searfus, Joseph E. Tull (1991) Deployable Seismic Verification System (DSVS): System Evaluation Center (SEC) Software Documentation. Version 1.0, Lawrence Livermore National Laboratory, M-281

Geiger (1910), apparently the first person with the idea of minimizing a sum of weighted squared residuals to estimate hypocentral coordinates.

Jackson, D. D. (1972), "Interpretation of Inaccurate, Insufficient and Inconsistent Data," *Geophys. J. R. Astr. Soc.* **28**, 97–109.

Jarpe, S. P., T. F. Hauk, and K. K. Nakanishi (1986), "Detection of Regional Earthquakes in Eastern North America using the RSTN in the Frequency Range 0.5 to 16 Hz," *Proceedings of the 1986 RSTN/ NORESS Research Symposium, Las Vegas, Nevada April 30–May 1, 1986*, Lawrence Livermore National Laboratory, Livermore, CA, Conf-8604196.

Jordan, T. H., and K. A. Sverdrup (1981), "Teleseismic Location Techniques and their Applications to Earthquake Clusters in the South-Central Pacific," *Bull. Seismol. Soc. Am.* **71**, 1105–1130.

Keller, G. R., R. B. Smith, L. W. Braile, R. Heaney, and D. H. Shurbet (1976), "Upper Crustal Structure of the Eastern Basin and Range, Northern Colorado Plateau, and Middle Rocky Mountains from Rayleigh-Wave Dispersion," *Bull. Seismol. Soc. Am.* 66, 869–876.

Lee, W. and J. Lahr (1971) HYPO71, a hypocentral location program; U.S. Geological Survey.

Menke, W. (1984), *Geophysical Data Analysis: Discrete Inverse Theory,* Harcourt Brace Jovanovich, Academic Press, Inc., 260 pp.

Nakanishi, K.K., S.R. Taylor, N.C. Burr, and P.W. Rodgers (1983) *Broadband Seismograms from Merged Mid- and Short-Period RSTN Data,* Lawrence Livermore National Laboratory, Livermore, CA, UCID-19884.

Owens, T. J., S. R. Taylor, and G. Zandt (1987), "Crustal Structure at Regional Seismic Test Network Stations Determined from Inversion of Broadband Teleseismic *P* Waveforms," *Bull. Seismol. Soc. Am.* **77**, 631–662.

Prodehl, C.(1970), "Seismic Refraction Study of Crustal Structure in the Western United States," *Bull. Geol. Soc. Am.* **81**, 2629–2646.

Taylor, S. R., and B. J. Qualheim (1983), *RSTN Site Descriptions*, Lawrence Livermore National Laboratory, Livermore, CA, UCID-19769.att, S. R., and T. C. Bache (1988), "Locating Events with a Sparse Network of Regional Arrays," *Bull. Seismol. Soc. Am.* 78, 780–798.

Acknowledgments

I would like to thank Steve Jarpe for significant help in data processing, Steve Jarpe and Keith Nakanishi for many helpful discussions and rewiew, Joe Tull for developing SAC, and Elaine Price for editoral revisions.

Table 1

RSTN Station Locations

Station	Latitude	Longitude
rssd	44.1204	104.0362
rsny	44.5483	074.5300
rson	50.8589	093.7022
rsnt	62.4797	114.5917
rscp	35.6000	085.5686

Table 2

Events for Study

yrdy hrmn	Latitude	Longitude	Depth	Mag	master
822672219:16.9	35.689	084.247	10.0	3.4	
822682317:05.4	35.210	092.230	04.0	3.5	
822760431:05.0	51.250	062.810	18.0	3.3	
822830435:42.1	49.950	063.530	18.0	2.6	
822871253:46.3	36.102	102.571	05.0	3.9	
822960549:07.1	65.710	090.370	18.0	2.9	
822961228:43.5	47.149	112.712	05.0	3.6	
822991531:32.6	47.000	066.600	05.0	3.6	
823190258:22.9	43.007	097.850	05.0	4.3	
823251635:28.3	35.190	092.230	03.0	3.4	
823261009:01.4	39.739	107.581	05.0	2.9	
823302002:04.3	46.269	111.988	05.0	3.4	
823381608:32.0	47.500	070.200	18.0	4.0	
823510547:41.0	43.700	074.210	06.0	2.0	
823511019:14.0	49.310	085.710	01.0	2.7	
823570706:39.3	42.720	081.480	09.0	2.9	
823600811:51.1	64.470	090.720	18.0	2.8	
823600846:22.2	64.990	086.610	18.0	3.0	
830060235:04.0	52.120	106.930	01.0	2.7	
830102131:27.0	46.820	078.840	18.0	3.3	
830170819:42.0	48.960	067.250	15.0	2.5	
830191735:51.0	49.110	066.980	18.0	4.1	
830190230:40.3	35.178	092.220	11.0	3.5	
830200916:45.0	48.720	083.450	18.0	3.0	
830201417:21.0	47.460	067.860	17.0	3.1	
830242115:01.1	43.966	073.358	11.0	2.4	
830261407:44.8	32.728	083.375	05.0	3.5	
830272209:35.0	36.054	083.619	15.0	2.6	
830361308:19.5	34.700	088.370	03.0	2.9	
830421546:56.0	48.990	068.300	18.0	3.6	
830431425:19.0	55.830	082.740	18.0	2.6	
830441344:44.0*	42.232	105.729	05.0	4.0	Churchill III
830470622:09.3	48.539	112.373	14.0	3.5	
830470714:07.4	45.927	111.497	05.0	3.7	
830472056:33.0	51.320	067.930	18.0	2.0	
830540851:27.0	36.192	089.604	09.0	3.6	
830612323:19.4	34.302	106.892	08.0	4.1	
830630632:18.6	44.214	099.409	05.0	4.4	
830730911:20.0*	50.820	074.900	18.0	3.7	Superior
830760725:56.6*	47.526	112.702	05.0	4.2	Churchill II
830790300:02.0	51.600	115.070	18.0	2.2	
830810455:04.0	51.640	115.030	05.0	3.6	

830811154:45.0	48.180	114.080	18.0	2.8	
830820141:29.2	47.930	114.000	18.0	3.2	
830821147:23.0	65.690	111.560	18.0	3.3	
830840247:11.1	35.345	082.462	09.0	3.2	
830930455:21.2	35.448	102.321	05.0	3.4	
831111702:29.0	53.520	075.580	02.0	2.5	
831142159:30.0	53.570	075.700	02.0	3.6	
831181035:03.2	54.000	081.000	18.0	2.9	
831200734:20.1	33.316	106.438	07.0	3.5	
831260614:46.9	42.955	102.198	05.0	3.3	
831331726:02.0	47.000	066.600	05.0	3.6	
831332340:57.0	47.000	066.600	05.0	4.1	
831350516:21.6*	38.770	089.570	09.0	4.3	Central
831360201:57.0	47.700	069.930	10.0	4.0	
831430149	46.550	080.830	01.0	2.6	
831490545:49.8	44.502	070.415	02.0	3.9	
831530630:23.0	47.450	070.220	10.0	3.4	
831542239:00.6	38.140	088.410	22.0	2.7	
831621347:58.0	47.000	066.600	05.0	3.4	
831671636:36.0	44.740	072.550	01.0	2.7	
831721358:28.5	35.190	092.180	04.0	2.5	
831730404:14.4	35.230	092.220	04.0	2.1	
831790805:49.0	47.050	066.690	05.0	3.3	
832241408:47.6	44.970	067.680	12.0	3.6	
832261908:30.7	38.359	107.402	05.0	3.4	
832291403:15.0*	38.472	082.772	12.0	3.5	Green2
832671657:45.7	40.789	108.837	05.0	4.1	
832801018:46.1*	43.938	074.258	13.0	5.1	Green1
832801039:38.5	43.952	074.258	08.0	3.5	
832840410:55.0	45.210	075.770	15.0	4.2	
832891940:50.8	30.243	093.393	05.0	3.8	
833051016:52.0	45.680	073.900	18.0	3.5	
833062003:58.8	43.418	110.921	05.0	3.5	
833100902:19.8	32.937	080.159	10.0	3.1	
833131353:12.9	43.716	110.200	05.0	3.6	
833191233:12.1	43.016	105.955	05.0	3.0	
833211532:18.0	47.000	066.600	05.0	3.7	
833231622:20.1	41.830	081.090	18.0	2.5	
833351654	38.710	090.840	10.0	3.2	
833381048:33.7	45.190	069.140	01.0	3.4	
833432052:10.5	33.183	092.704	05.0	3.0	
833542252:23.7	43.294	110.767	05.0	4.5	
833542321:52.5	43.268	110.826	05.0	3.5	
833551504:44.0	45.220	073.980	11.0	3.1	
833552025:22.2	46.939	113.542	05.0	3.0	
833561856:03.9	43.224	110.802	05.0	3.4	
833621224:21.0	47.010	076.330	18.0	3.4	
840840521:51.8	43.329	110.782	05.0	2.8	
841080444:43.8	38.381	088.435	20.0	3.4	
841241825:35.4	39.291	107.229	05.0	2.5	
841270200:56.6	39.335	107.249	05.0	2.1	
841380911:20.2	39.337	107.245	05.0	2.4	

* master events for table 11

Table 3

Weights based upon signal to noise ratio

signal to noise ratio	weight
$10:1 < SNR$	0
$5:1 < SNR \leq 10:1$	1
$2:1 < SNR \leq 5:1$	2
$1:1 < SNR \leq 2:1$	3
$SNR \leq 1:1$	4

Table 4

Minimum Phase Information to Locate Epicenter

No. of Stats.	1 AT	2 AT	BAZ	1 AT & BAZ	2AT & BAZ
one stat					X
two stat	X			X	
two stat			XX		
three stat	XXX				

Table 5

Number of Events Located

Magnitude	Total	No. Located
5.0-5.4	1	1
4.5-4.9	1	1
4.0-4.4	12	12
3.5-3.9	26	25
3.0-3.4	27	23
2.5-2.9	21	16
2.0-2.4	5	2

Table 6

Velocity Model I

Top of Layer (km)	Vp	Vs	V_{Lg}
0.0	6.0	3.5	3.5
15.0	6.8	4.0	
40.0	8.0	4.7	

Table 7

Travel Time Residuals Squared

Station	Pg	N	Pn	N	Sn	N	Lg	N	BAZ	N
RSON	532.0	35	132.7	22	432.8	27	276.2	34	4707.0	13
RSNT	532.0	35*	120.0	09	060.0	08	009.3	05	11255.0	29
RSSD	201.8	13	130.9	19	095.1	14	338.0	31	5143.0	08
RSCP	175.1	12	089.8	15	204.2	16	257.6	29	11255.0**	29
RSNY	156.5	10	227.5	17	160.1	21	546.2	51	1405.0	11

* no data available, obtained from RSON

** no data available, obtained from RSNT

Table 8

Location Summary - Average Velocity Structure

yrdayhrmn sec	Latitude	Longitude	Mag	St	Variance	Loc error	95% error
822672219:18.4	35.985	84.411	3.4	3	4.29	36.1	111.6
822682317:07.0	35.386	92.055	3.5	4	5.49	25.1	77.9
822760431:11.6	49.883	59.619	3.3	2	5.10	272.5	1557.1
822871253:53.2	36.294	101.926	3.9	4	5.33	61.7	73.4
822960549:15.6	73.076	117.123	2.9	1	10.59	1316.6	2545.2
822961228:51.9	47.135	112.067	3.6	2	6.70	49.0	106.7
822991531:35.8	46.345	66.563	3.6	2	4.98	72.9	1528.5
823190258:18.3	43.043	97.585	4.3	4	5.76	22.0	29.5
823251635:28.8	35.420	92.083	3.4	3	6.57	28.7	115.3
823261008:52.3	41.951	111.403	2.9	1	10.83	405.0	1038.7
823302002:03.9	46.692	111.726	3.4	3	4.40	51.1	97.6
823381608:33.2	47.869	71.201	4.0	5	6.17	85.6	105.6
823510547:10.5	51.343	67.170	2.0	1	11.54	1000.6	2415.2
823511019:39.5	52.994	87.633	2.7	1	10.38	431.4	979.2
823570706:14.2	44.066	83.282	2.9	1	8.20	208.9	658.0
823600811:45.2	66.945	93.038	2.8	1	10.59	296.0	2303.5
823600846:51.9	52.864	103.690	2.8	1	8.45	1490.4	1979.8
830060234:52.9	51.741	107.245	2.7	3	8.65	47.4	106.0
830102131:24.0	47.408	78.819	3.3	2	7.37	65.4	598.2
830170820:57.2	42.263	69.077	2.5	1	2.93	757.7	455.9
830191735:52.7	49.348	68.231	4.1	4	4.62	94.9	88.8
830190230:45.5	35.681	91.862	3.5	4	5.36	64.6	88.3
830200916:47.9	49.012	82.898	3.1	2	6.36	51.9	569.7
830201417:24.5	47.451	68.307	3.1	1	4.99	33.8	802.6
830261407:46.4	33.082	83.506	3.5	3	2.54	41.1	43.1
830272209:34.3	37.172	85.042	2.6	1	4.98	177.6	262.3
830361308:16.5	34.707	88.345	2.9	2	3.18	3.1	78.5
830421546:57.0	49.227	69.078	3.6	3	4.10	62.6	95.0
830431425:53.6	53.852	84.890	2.6	1	7.65	260.0	1218.6
830441344:35.8*	42.361	106.355	4.0	5	5.90	53.6	75.9
830470622:03.1	48.622	112.426	3.5	3	7.10	9.8	109.9
830470714:09.2	46.302	111.201	3.7	3	4.91	47.6	90.4
830540851:34.0	37.091	88.895	3.6	4	5.48	118.1	93.0
830612323:33.9	34.679	106.159	4.1	3	4.56	79.2	70.0
830630632:12.5	44.132	99.096	4.4	4	5.93	26.6	26.0
830730911:23.4*	50.460	75.901	3.6	5	5.39	81.4	100.1
830760726:04.6*	47.430	111.759	4.2	3	6.31	71.7	105.3
830810454:55.8	51.567	115.424	3.6	2	5.41	28.5	101.1
830820141:55.7	47.381	112.325	3.2	1	5.29	139.8	1054.2
830840247:04.7	35.656	82.281	3.2	2	2.73	38.2	81.6
830930455:20.7	35.455	102.258	3.4	3	5.66	6.2	94.2
831142159:37.5	53.155	75.910	3.6	2	6.16	48.3	163.4
831181035:17.0	52.802	81.891	2.9	2	4.83	146.0	111.9
831200734:22.9	33.469	106.600	3.5	3	2.78	22.6	46.9
831260614:54.2	43.447	102.489	3.3	2	6.58	59.5	98.3
831331726:16.0	47.307	67.759	3.6	2	6.39	94.3	98.4
831332340:46.7	46.669	65.958	4.1	5	7.41	61.3	133.6
831350516:19.0*	38.989	89.440	4.3	4	6.38	26.7	36.5
831360202:01.0	47.832	70.757	4.0	5	6.13	63.7	100.5
831490545:58.4	44.792	71.448	3.9	5	5.57	88.0	89.0
831530630:44.5	47.240	72.240	3.4	3	5.86	154.4	102.5

831621348:04.1	47.312	67.296	3.4	2	3.91	63.2	78.0
831671636:36.3	44.601	72.582	2.6	1	4.10	15.9	971.8
831721358:18.3	36.209	91.378	2.5	2	5.21	134.3	185.7
831730404:37.1	34.145	88.115	2.1	1	5.18	394.6	265.4
831790805:52.4	47.469	67.266	3.3	2	3.79	63.8	76.7
832241408:49.6	45.741	68.153	3.6	2	4.66	93.3	82.0
832261908:30.5	40.514	111.636	3.4	1	10.89	435.7	1290.4
832291403:17.6*	38.988	82.920	3.5	3	5.48	58.7	74.8
832671657:44.3	39.477	106.540	4.1	2	4.80	243.9	694.7
832801018:45.3*	44.112	75.040	5.1	5	6.17	65.6	65.4
832801039:44.4	44.174	74.908	5.1	3	7.84	58.3	77.7
832840410:49.5	45.132	76.168	4.2	4	4.86	32.5	47.0
832891940:46.5	29.861	93.461	3.8	3	5.19	42.8	78.0
833051016:46.7	45.939	74.246	3.5	2	3.42	39.4	136.1
833100902:26.0	33.532	80.278	3.1	2	3.61	66.9	64.8
833131353:13.6	44.244	109.584	3.6	2	4.49	76.7	110.8
833191233:13.6	43.811	106.422	3.0	2	2.30	96.0	75.8
833211532:24.2	47.398	67.343	3.7	2	3.86	71.6	75.2
833381048:34.7	45.929	69.622	3.4	2	4.62	90.3	112.1
833542252:26.6	43.906	110.507	4.5	3	6.14	71.2	118.1
833542321:58.9	43.822	110.302	3.5	2	5.73	74.7	371.2
833551504:42.7	44.655	73.412	3.1	1	3.40	77.2	237.3
833552025:31.2	46.590	113.025	3.0	2	8.07	55.3	230.9
833561856:11.3	44.238	110.346	3.4	2	5.71	118.5	325.2
833621224:14.7	47.028	76.551	3.4	3	3.55	17.0	60.9
840840521:54.4	45.222	110.314	2.8	1	5.29	213.6	760.6
841080444:39.1	38.548	88.195	3.4	4	5.53	28.0	46.5
841241825:30.0	42.250	110.918	2.5	1	5.29	452.5	678.3
841270200:58.1	42.056	110.590	2.1	1	5.29	413.3	681.6
841380911:22.5	39.615	107.361	2.4	1	5.29	32.4	860.5

*master events

Table 9

North American Crustal Models

Province	upper crust	V_p	V_s		lower crust	V_p	V_s		mantle	V_p	V_s	
Churchill	12 km	5.9	3.6	(km/sec)	25 km	6.9	4.0	(km/sec)		7.9	4.6	(km/sec)
Superior	14	6.3	3.7		20	6.9	4.0			8.1	4.7	
Central	20	6.1	3.6		19	6.9	4.0			8.1	4.6	
Grenville	21	6.3	3.6		18	6.9	3.9			8.2	4.7	

Table 10

Station Corrections

	RSON					RSNT					RSSD				
Province	Pg	Pn	Sn	Lg	Bz	Pg	Pn	Sn	Lg	Bz	Pg	Pn	Sn	Lg	Bz
Churchill I	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Churchill II	-3.4	-3.6	2.0	1.9	0.0	0.0	0.0	0.0	0.0	0.0	4.3	3.1	0.0	-3.6	0.0
Churchill III	0.0	-8.1	-0.1	1.2	0.0	0.0	0.0	7.9	1.6	0.0	-2.9	0.0	0.0	5.3	0.0
Superior	0.0	-0.3	5.1	3.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	-8.1	19.8	0.0
Central	0.0	-0.2	-3.0	1.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	-0.2	-1.8	0.6	0.0
Grenville I	0.0	-1.6	3.2	-3.1	0.0	0.0	1.2	-3.6	0.0	0.0	0.0	-4.6	17.5	5.4	0.0
Grenville II	0.0	-6.8	-4.0	-0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.5	4.1	0.0

	RSCP					RSNY				
Province	Pg	Pn	Sn	Lg	Bz	Pg	Pn	Sn	Lg	Bz
Churchill I	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Churchill II	0.0	-0.1	0.0	-0.1	0.0	0.0	0.0	0.0	0.0	0.0
Churchill III	0.0	-1.8	1.9	9.3	0.0	0.0	0.0	0.0	5.3	0.0
Superior	0.0	0.0	0.0	6.9	0.0	0.0	3.6	0.8	-3.2	0.0
Central	-3.5	2.4	3.7	-2.7	0.0	0.0	2.0	1.1	-2.4	0.0
Grenville I	-2.5	1.7	5.1	-1.6	0.0	2.7	1.0	-0.5	3.7	0.0
Grenville II	0.0	0.0	0.0	0.0	0.0	0.0	-2.3	2.4	1.7	0.0

Table 11

Travel Time Residuals Squared

Station	Pg	N	Pn	N	Sn	N	Lg	N	BAZ*	N
Churchill I										
RSON	000.0	00	027.0	10	000.0	00	027.9	08	0661.0	04
RSNT	000.0	00	109.8	04	000.0	00	106.4	05	0000.0	00
RSSD	268.2	09	023.9	08	005.2	03	098.5	11	0321.0	11
RSCP	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSNY	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
Churchill II										
RSON	000.0	00	027.0	10	000.0	00	027.9	08	0661.0	04
RSNT	000.0	00	109.8	04	000.0	00	106.4	05	0000.0	00
RSSD	268.2	09	023.9	08	005.2	03	098.5	11	0321.0	11
RSCP	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSNY	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
Churchill III										
RSON	000.0	00	063.2	05	056.2	04	037.7	04	0000.0	00
RSNT	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSSD	052.5	08	013.1	04	054.0	04	168.2	08	0000.0	00
RSCP	000.0	00	005.5	03	000.0	00	148.4	04	0000.0	00
RSNY	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
Superior										
RSON	000.0	00	000.0	00	000.0	00	000.0	00	0058.0	02
RSNT	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSSD	000.0	00	000.0	00	000.0	00	000.0	00	0032.4	01
RSCP	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSNY	000.0	00	000.0	00	000.0	00	005.0	03	0019.0	02
Central										
RSON	000.0	00	067.9	04	155.8	05	043.7	05	0000.0	00
RSNT	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSSD	000.0	00	016.4	04	014.8	05	080.5	07	0000.0	00
RSCP	048.8	07	038.5	04	016.7	06	012.5	04	0000.0	00
RSNY	000.0	00	000.0	00	000.0	00	062.5	04	0000.0	00
Grenville I										
RSON	000.0	00	051.2	05	035.6	04	063.5	13	0162.8	06
RSNT	000.0	00	000.0	00	128.0	04	000.0	00	0000.0	00
RSSD	000.0	00	000.0	00	103.2	03	108.5	04	0000.0	00
RSCP	000.0	00	022.0	04	000.0	00	197.2	06	0000.0	00
RSNY	386.8	09	110.3	12	098.5	11	074.5	11	0016.3	04
Grenville II										
RSON	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSNT	000.0	00	000.0	00	000.0	00	000.0	00	0000.0	00
RSSD	000.0	00	000.0	00	000.0	00	009.2	03	0000.0	00
RSCP	010.2	03	000.0	00	000.0	00	007.2	05	0000.0	00
RSNY	000.0	00	000.0	00	000.0	00	004.4	02	0000.0	00

*backazimuth residuals have been weighted as discussed following eq. 30.

Table 12

Location Summary - Regional Velocity Structure

yrdayhrmn sec	Latitude	Longitude	Mag	St	Variance	Loc error	95% error
822672219:16.5	35.558	84.192	3.4	3	4.46	15.4	88.2
822682317:07.1	34.981	92.239	3.5	4	7.61	25.4	95.1
822760431:07.5	49.962	59.214	3.3	2	6.11	292.1	1800.3
822871253:47.8	35.977	102.290	3.9	4	5.68	28.9	74.0
822960549:18.9	72.781	117.012	2.9	1	14.15	1299.7	3211.6
822961228:36.6	47.059	113.211	3.6	2	9.73	39.2	138.7
822991531:34.7	46.350	66.486	3.6	2	6.22	72.8	1804.4
823190258:19.4	42.924	97.790	4.3	4	6.69	10.5	25.2
823251635:29.2	35.037	92.199	3.4	3	6.97	17.2	120.5
823261008:53.8	41.970	111.349	2.9	1	19.34	402.7	1539.9
823302001:58.9	46.135	112.312	3.4	3	7.50	29.0	127.1
823381608:38.9	47.309	70.907	4.0	5	16.15	57.4	178.3
823510547:32.1	51.492	68.845	2.0	1	9.64	955.1	2918.9
823511019:39.5	52.994	87.633	2.7	1	10.38	431.4	979.1
823570706:22.5	44.109	82.811	2.9	1	21.52	188.2	1172.6
823600811:48.4	66.867	93.726	2.8	1	14.15	301.1	2899.4
823600846:32.0	52.162	103.116	2.8	1	12.55	1546.2	2516.2
830060234:51.5	52.001	107.476	2.7	3	12.76	39.8	142.7
830102131:22.6	46.563	79.207	3.3	2	5.34	40.1	1569.3
830170820:52.9	42.184	68.905	2.5	1	4.98	764.0	689.6
830191735:58.4	48.820	67.718	4.1	4	11.30	62.9	140.5
830190230:46.9	35.367	92.001	3.5	4	7.26	28.8	103.0
830200916:51.5	48.629	83.378	3.1	2	4.78	11.6	926.6
830201417:24.1	47.456	68.295	3.1	1	6.36	32.8	960.3
830261407:49.3	32.987	83.377	3.5	3	1.62	28.7	33.8
830272209:35.7	37.172	85.042	2.6	1	1.90	177.6	186.4
830361308:20.0	34.385	88.216	2.9	2	3.21	37.7	82.1
830421546:57.6	48.754	68.403	3.6	3	5.12	27.3	104.6
830431425:11.4	54.536	82.326	2.6	1	7.65	146.5	1434.5
830441344:46.6*	42.155	105.483	4.0	5	4.36	22.1	66.3
830470622:01.2	48.468	112.836	3.5	3	11.01	35.1	152.2
830470714:04.7	45.711	111.713	3.7	3	8.90	29.2	134.7
830540851:37.6	36.915	88.945	3.6	4	7.96	99.5	107.0
830612323:17.8	34.115	107.025	4.1	3	5.69	24.1	76.5
830630632:14.4	44.055	99.315	4.4	4	6.84	19.2	18.8
830730911:12.0*	51.221	74.515	3.6	4	4.74	52.1	96.1
830760725:53.1*	47.367	112.879	4.2	3	9.00	22.3	133.4
830810454:50.3	51.966	116.142	3.6	2	10.05	84.9	154.6
830820141:24.5	47.952	114.082	3.2	1	13.96	6.2	2127.2
830840247:08.9	35.559	82.356	3.2	2	1.09	25.5	20.7
830930455:15.9	35.193	102.565	3.4	3	5.60	36.0	87.6
831142159:24.6	54.003	74.516	3.6	2	5.55	91.7	161.4
831181035:11.2	53.911	81.555	2.9	2	5.07	37.8	105.5
831200734:02.5	32.646	107.448	3.5	3	4.70	120.0	64.7
831260615:00.9	44.989	104.698	3.3	2	5.32	302.1	145.7
831331726:19.2	46.590	67.413	3.6	2	14.88	77.0	156.1
831332340:58.4	46.940	66.763	4.1	5	6.96	14.2	107.0
831350516:22.8*	38.897	89.461	4.3	4	7.57	16.9	38.8
831360201:56.9	47.561	69.992	4.0	5	15.57	16.2	161.9
831490546:00.0	44.601	70.949	3.9	5	16.14	43.9	147.5
831530630:32.8	47.166	70.959	3.4	3	5.26	64.2	97.6

831621348:02.9	46.515	66.763	3.4	2	4.81	55.3	86.2
831671636:37.3	44.598	72.739	2.6	1	5.27	21.8	1162.7
831721358:16.5	35.405	91.674	2.5	2	3.11	51.8	106.4
831730404:18.9	33.707	88.849	2.1	1	2.07	352.5	318.5
831790805:51.7	46.705	66.740	3.3	2	4.67	38.6	79.0
832241408:51.4	44.647	68.095	3.6	2	10.58	48.6	291.4
832261908:34.0	40.613	111.451	3.4	1	16.42	428.6	1699.2
832291403:21.3*	38.812	82.610	3.5	3	5.86	40.3	80.8
832671657:46.5	40.423	108.387	4.1	2	5.71	55.6	1713.1
832801018:44.8*	44.028	74.313	5.1	5	18.32	11.0	167.3
832801039:41.8	44.028	74.160	5.1	3	9.54	12.8	137.5
832840410:51.2	45.216	75.776	4.2	4	12.23	0.0	98.5
832891940:47.2	29.734	93.591	3.8	3	6.99	59.5	90.0
833051016:45.8	45.741	73.545	3.5	2	11.08	28.5	163.0
833100902:25.0	33.204	80.309	3.1	2	4.04	32.6	63.5
833131353:01.5	43.559	110.247	3.6	2	8.07	17.9	177.6
833191233:17.9	43.278	105.917	3.0	2	2.44	29.3	107.2
833211532:22.3	46.631	66.754	3.7	2	4.73	42.7	85.5
833381048:35.8	45.020	69.423	3.4	2	10.34	29.2	200.4
833542252:13.4	42.894	111.155	4.5	3	9.90	54.6	147.2
833542321:42.4	42.151	110.700	3.5	2	8.97	124.5	742.0
833551504:42.1	44.645	73.517	3.1	1	19.89	73.6	615.4
833552025:18.1	46.653	114.104	3.0	2	10.32	53.4	277.2
833561855:55.3	42.720	110.971	3.4	2	9.12	57.7	569.8
833621224:18.9	47.053	75.850	3.4	3	11.71	36.7	140.7
840840521:27.7	45.435	111.865	2.8	1	13.96	249.4	1709.6
841080444:44.1	38.460	88.268	3.4	4	6.33	17.0	39.3
841241825:34.0	42.307	110.734	2.5	1	2.53	446.6	506.9
841270201:02.4	42.126	110.393	2.1	1	2.53	408.0	508.3
841380911:27.0	39.766	107.258	2.4	1	2.53	47.6	644.1

*master events

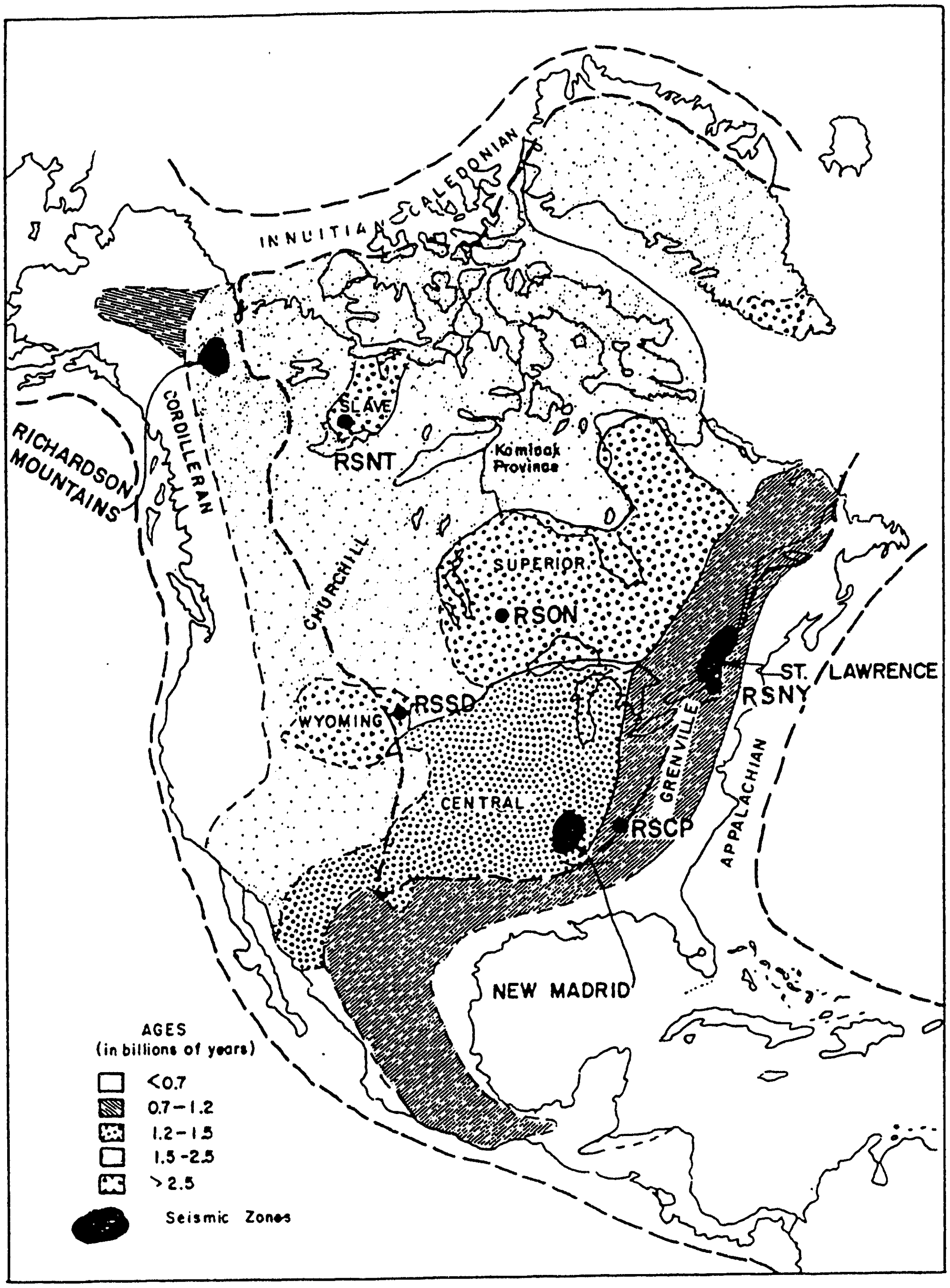
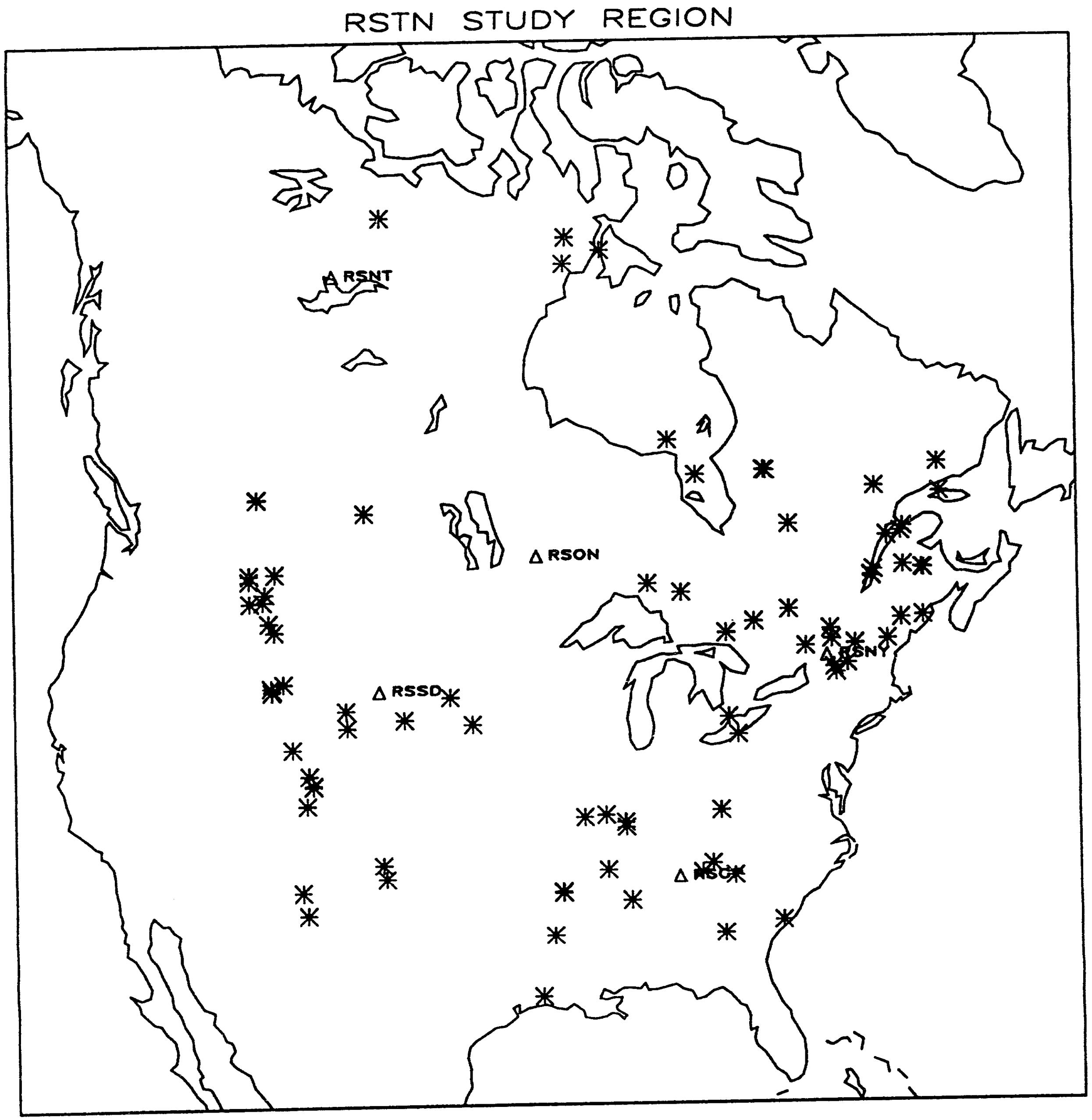


Figure 1. Locations of the RSTN stations, and major crustal provinces in North America (from Owens et. al., 1987). Major Precambrian regions are shown but excluding anorogenic provinces. Inner and outer boundaries of Phanerozoic belts are shown with the heavy dashed lines.



:/rstn/figures f001.sgf

Thu Nov 9 10:48:17 1989

Figure 2. Locations of 94 events studied obtained by local networks and listed in Table 2. The events used in this analysis were randomly selected, but an even distribution in location and magnitude was attempted.

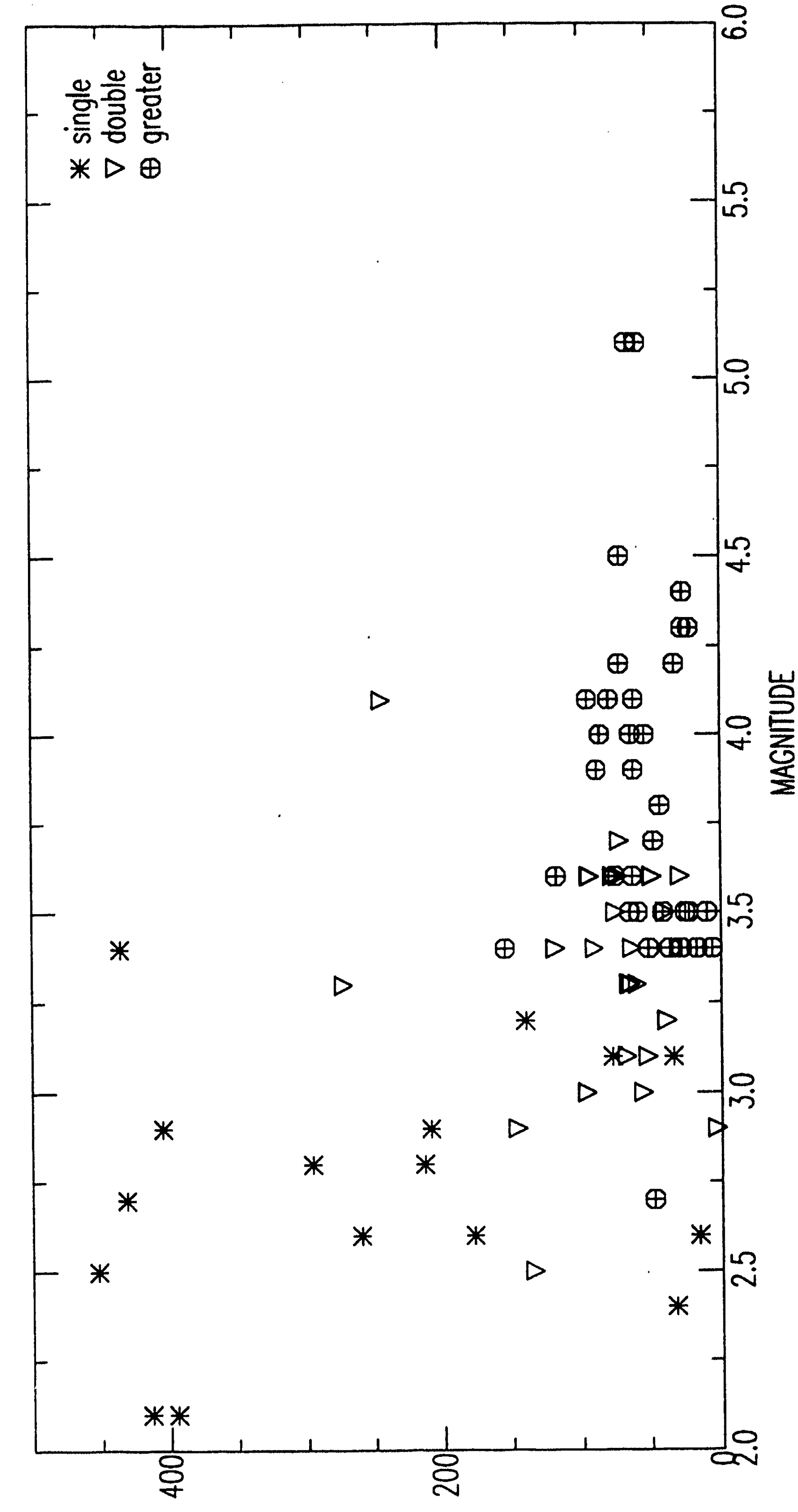


Figure 3. Location error vs. magnitude for events located with the average crustal model. events with location error grater than 500 km are not shown. Symbols indicate how many stations were used in the location: single (one), double (two), and greater (three or more).

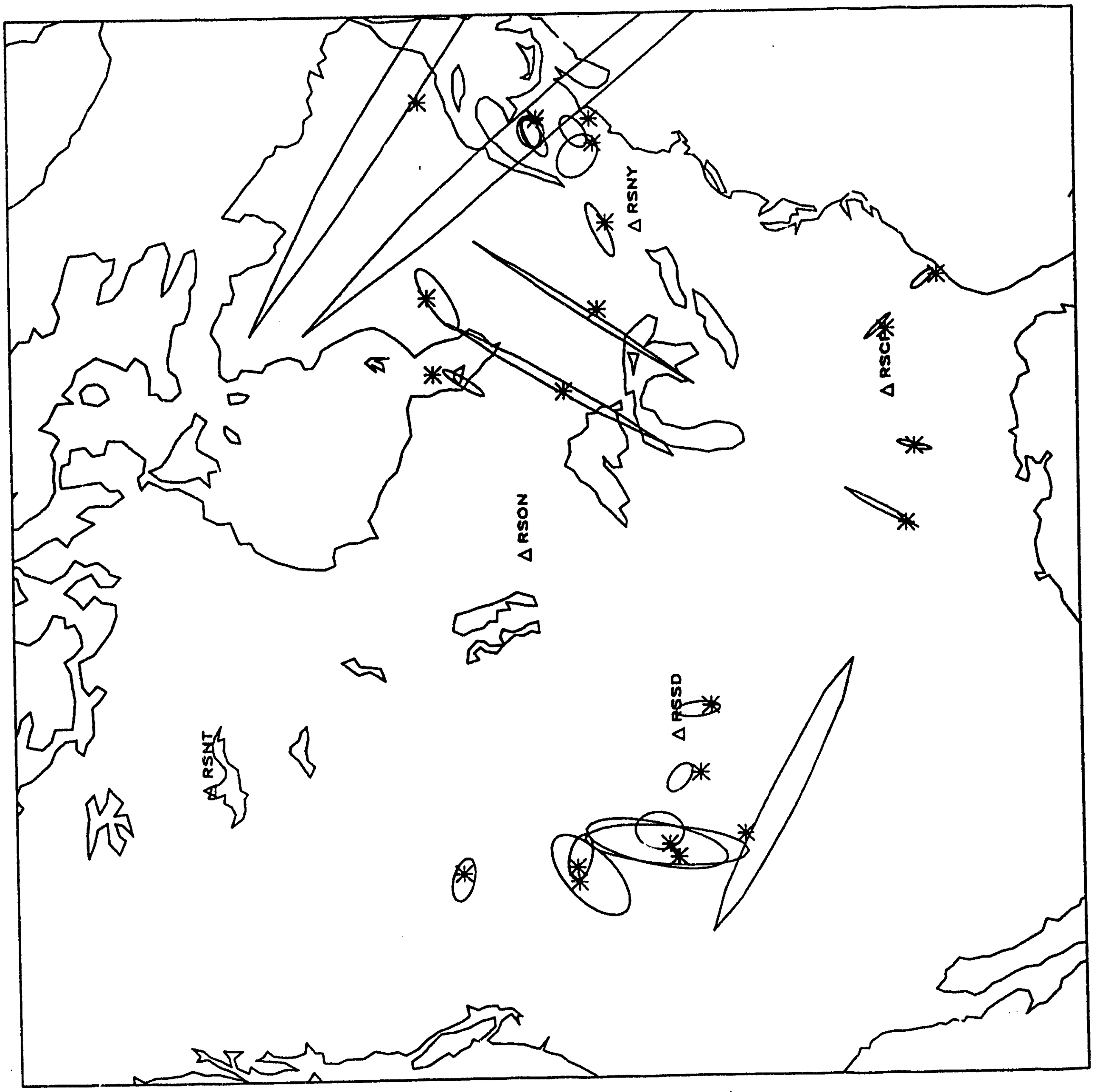


Figure 4. Ninety-five percent confidence error ellipses for events located with two stations and with the average crustal model. Also shown is the location of the events as determined by local networks. Seventeen of the twenty-six error ellipses include the location of the event.

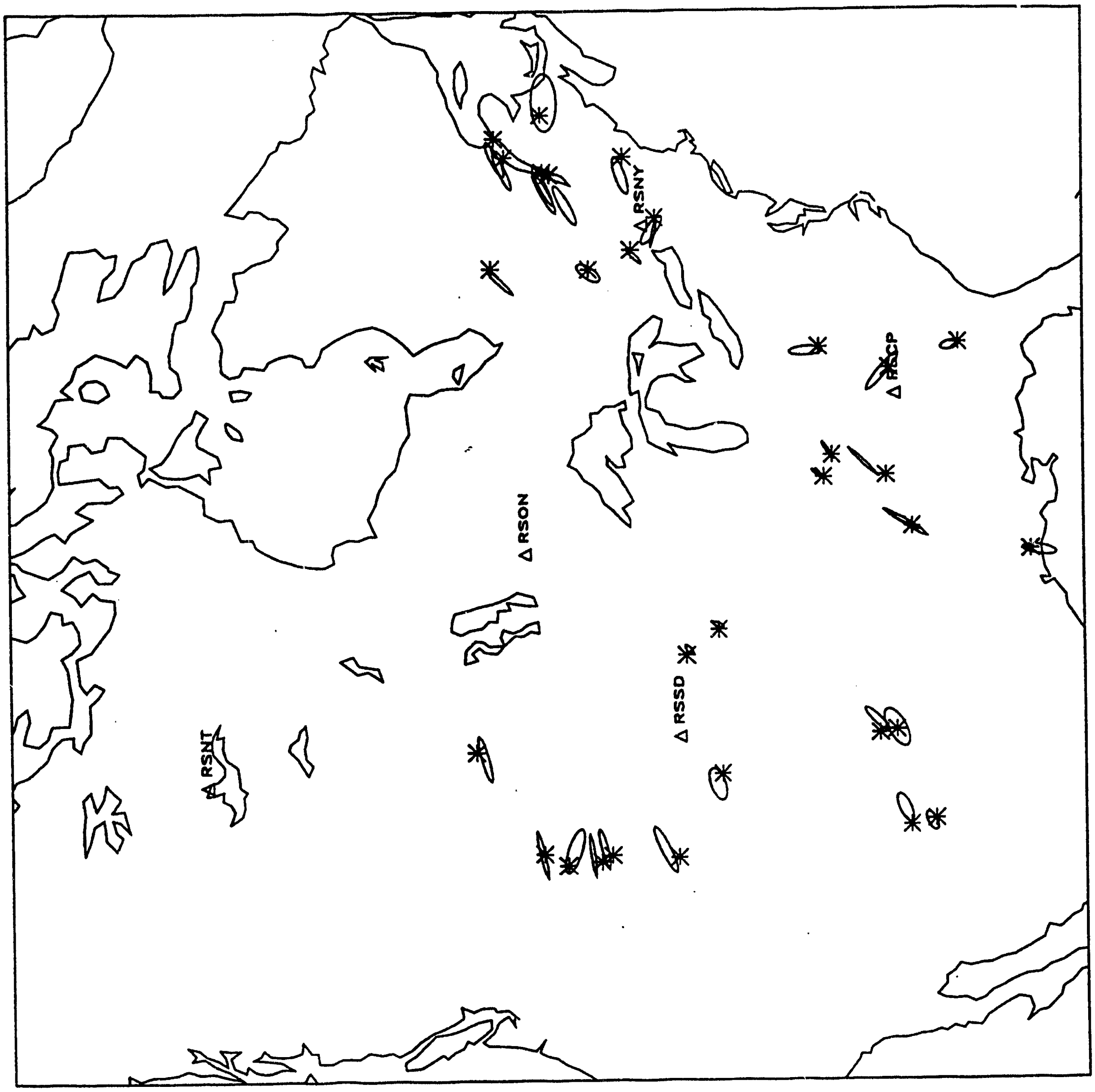


Figure 5. Ninety-five percent error ellipses for events located with three or more stations and with the average crustal model. Also shown is the location of the events as determined by local networks. Sixteen of the thirty-three error ellipses include the location of the event.

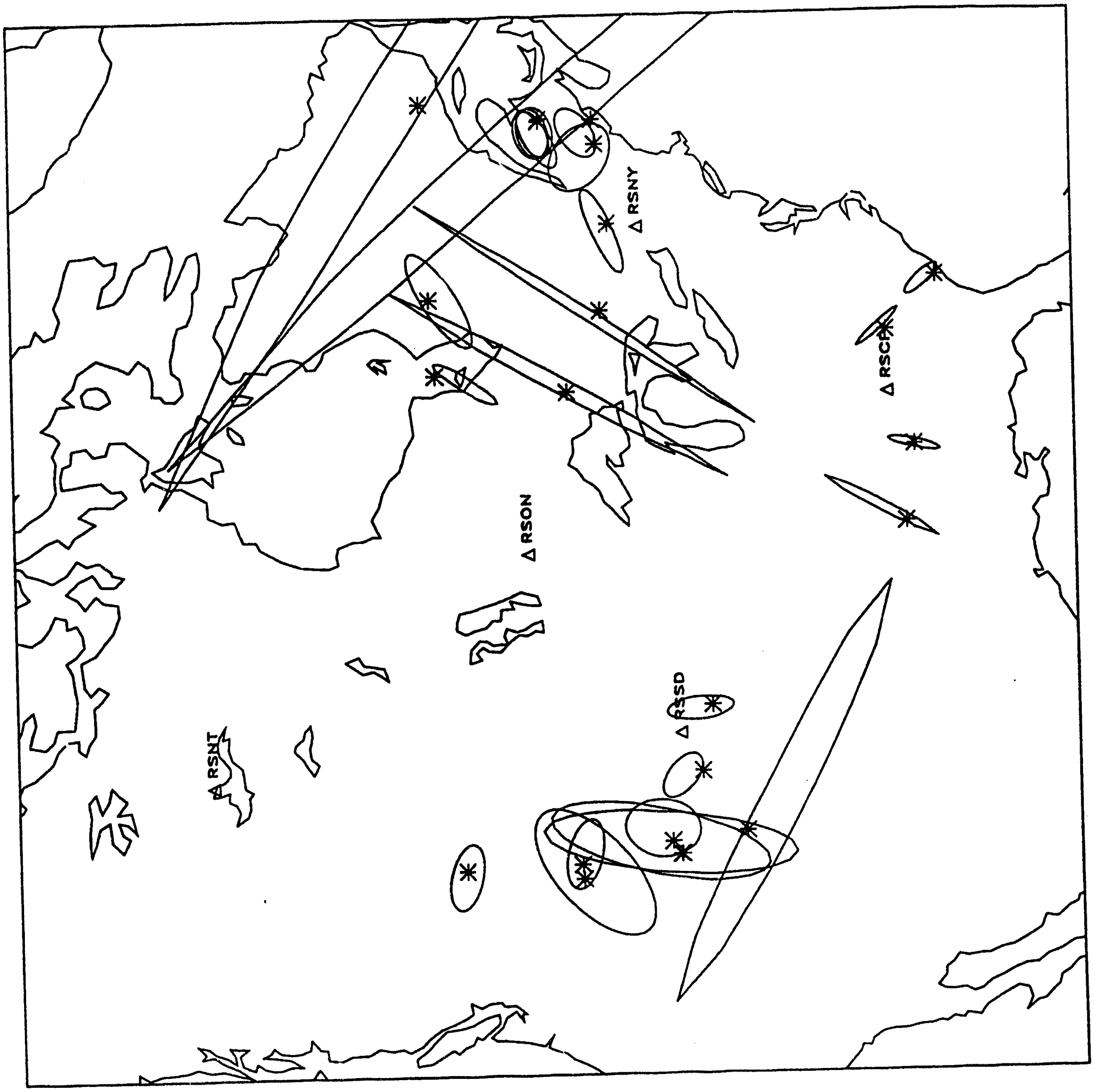


Figure 6. Ninety-nine percent confidence error ellipses for events located with two stations and with the average crustal model. Also shown is the location of the events as determined by local networks. Twenty of the twenty-six error ellipses include the location of the event.

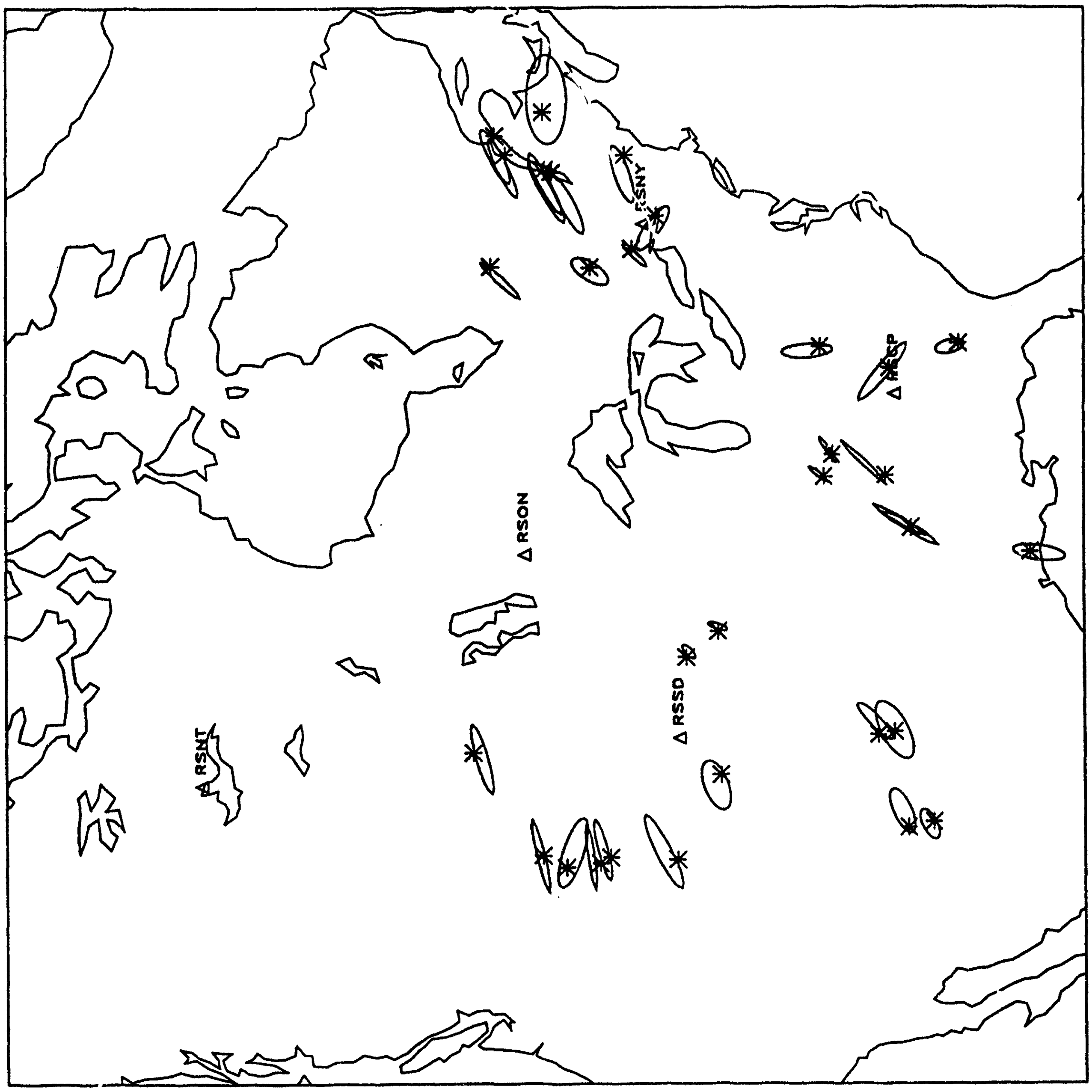


Figure 7. Ninety-nine percent error ellipses for events located with three or more stations and with the average crustal model. Also shown is the location of the events as determined by local networks. Twenty-four of the thirty-three error ellipses include the location of the event.

REGIONAL VELOCITY MODELS

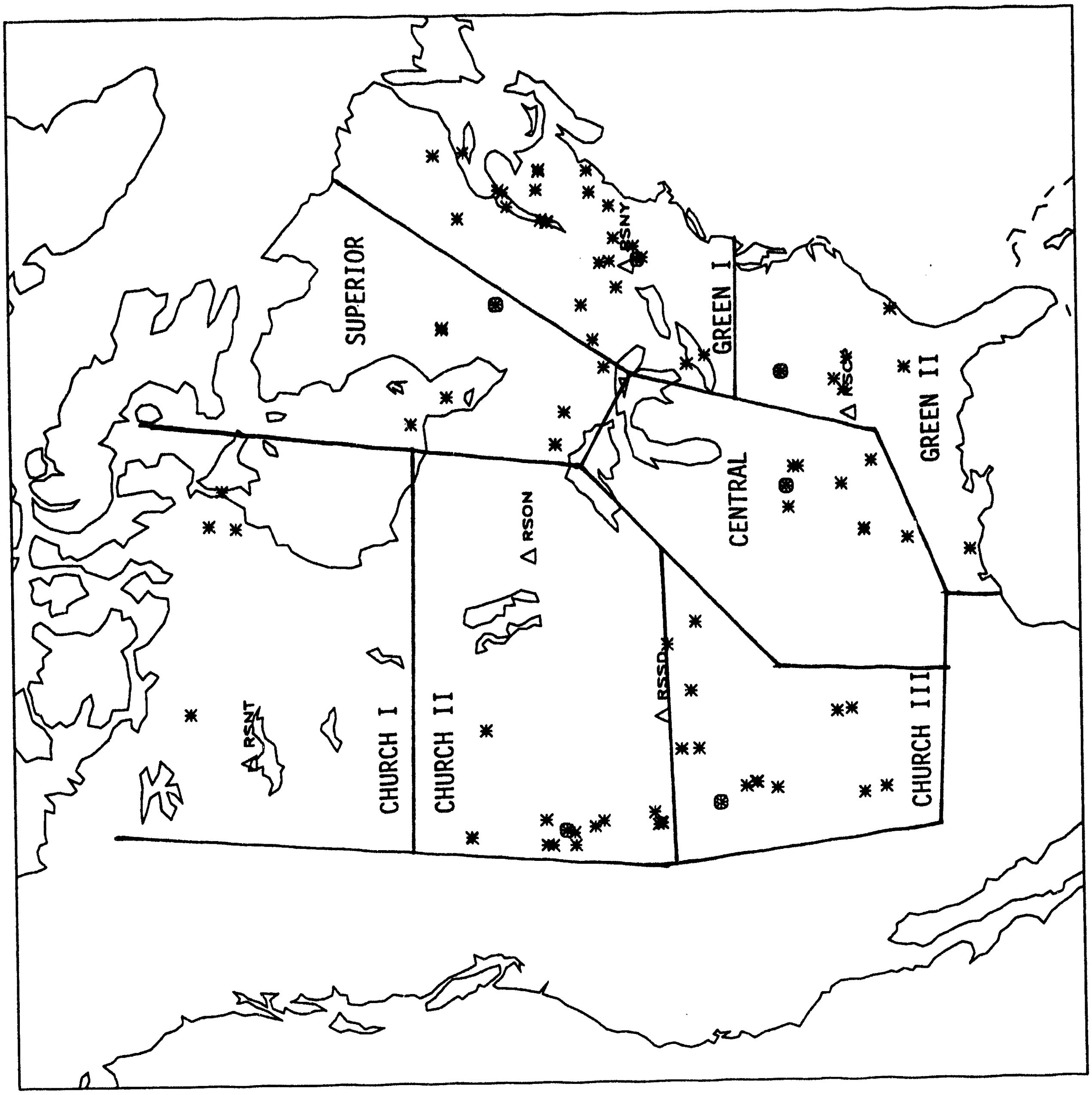


Figure 8. The study area divided into regions that roughly correspond to the provinces shown in Figure 1, along with event locations listed in Table 2 and station locations. The Greenville and Churchill provinces have been further divided into sub-regions for purposes of using master events in location calculations. Events located within each region, or subregion, are located with the same velocity model and station corrections. The velocity models for the provinces are listed in Table 9, and the station corrections are listed in Table 10. Master events used to obtain station corrections are shown by a different symbol. No master event was used for the Church I region.

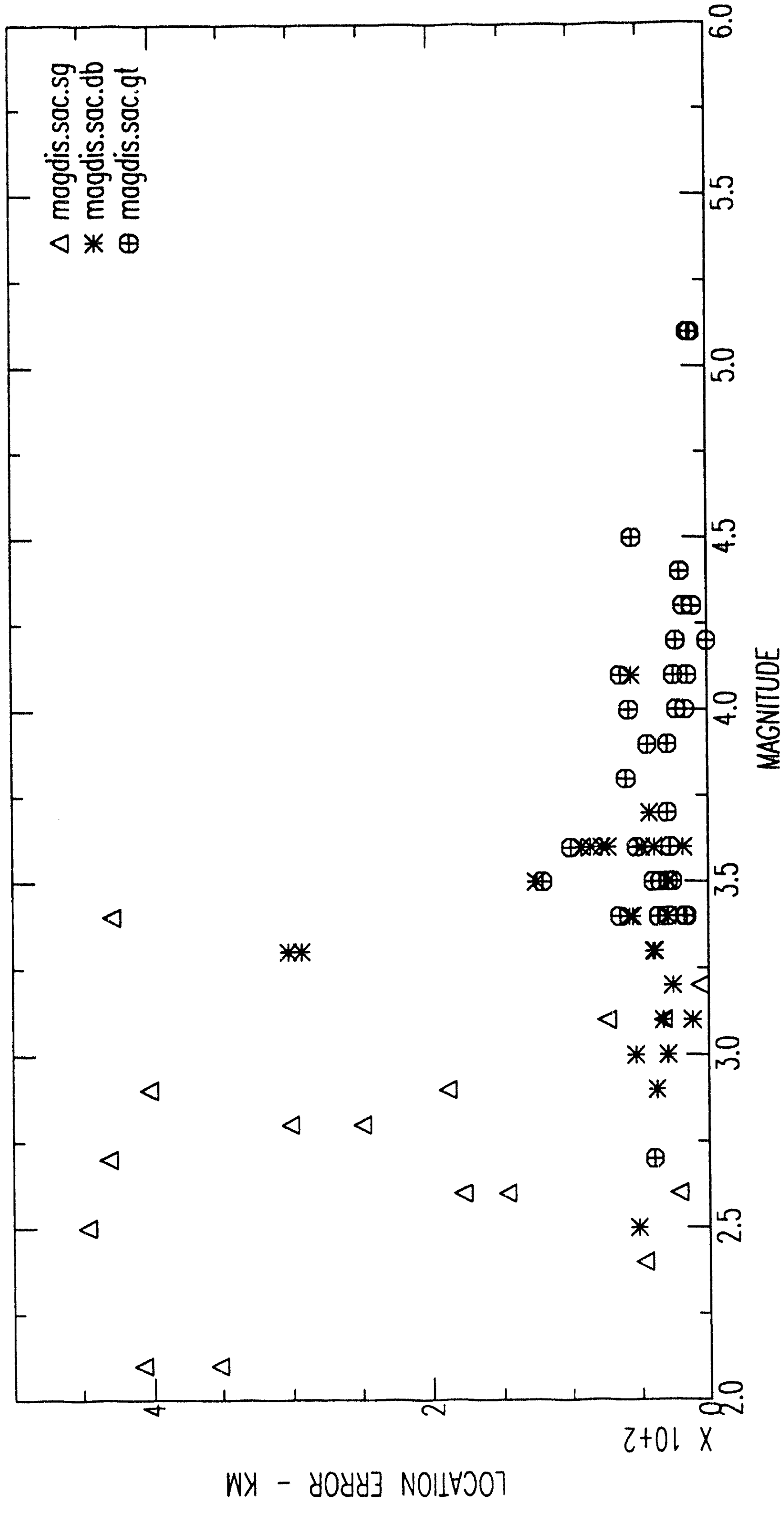


Figure 9. Location error as a function of magnitude.

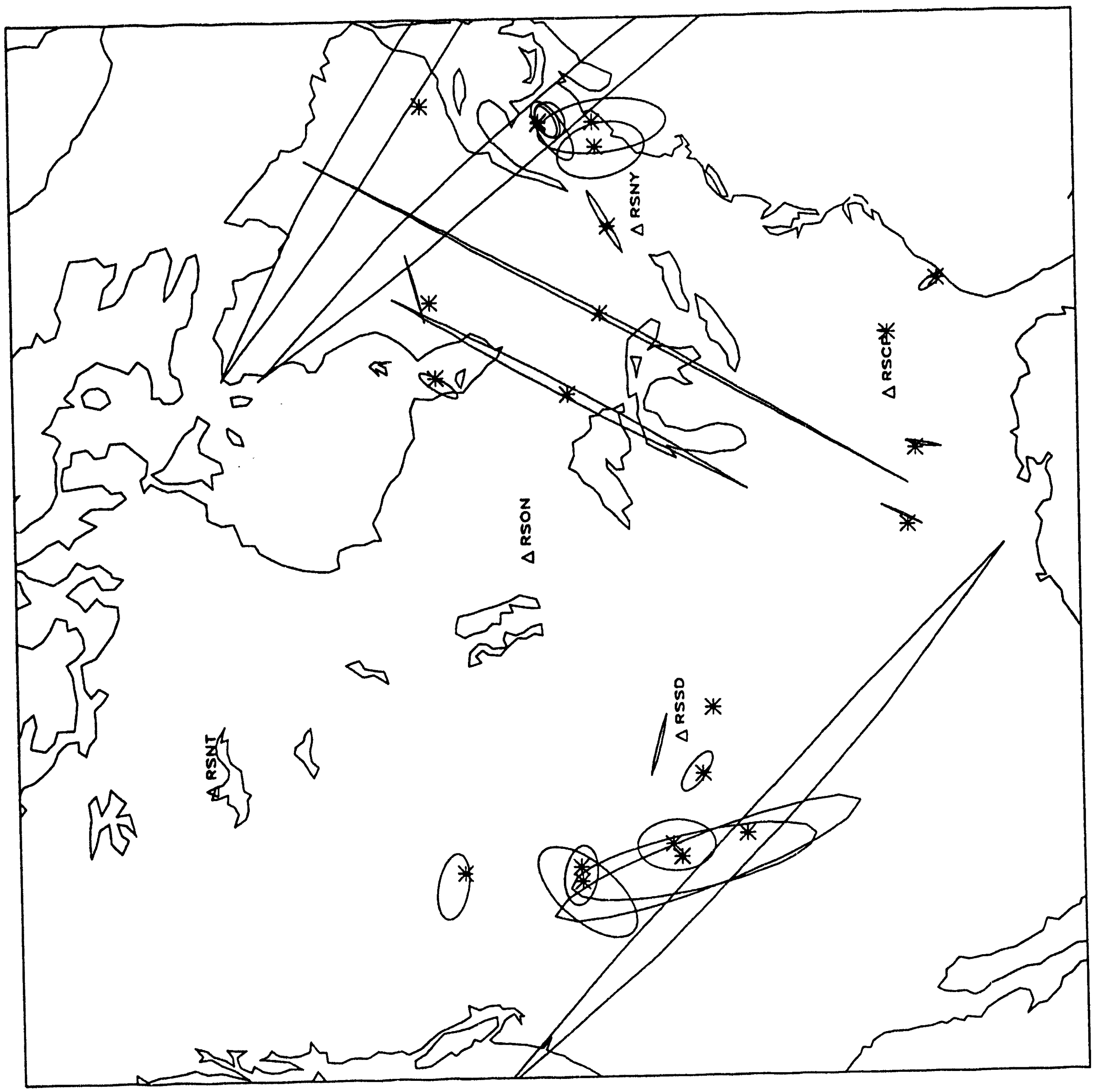


Figure 10. Location error ellipses with 95% confidence and the actual location of the event as listed in Table 2, and for events located with two stations. Twenty of the 26 ellipses include the actual location of the event.

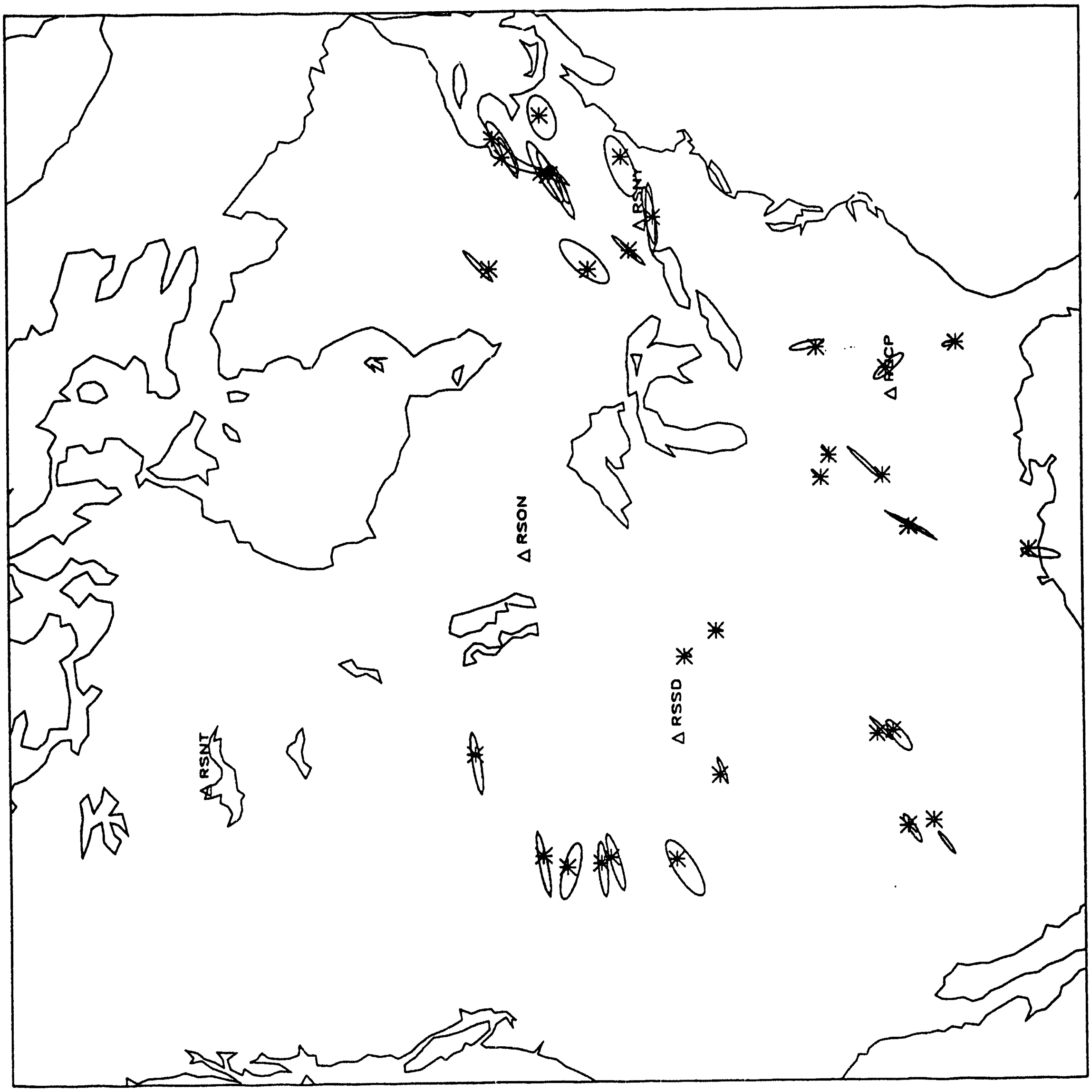


Figure 11. Location error ellipses with 95% confidence and the actual location of the event as listed in Table 2, and for events located with three or more stations. Twenty-eight of the 33 ellipses include the actual location of the event.

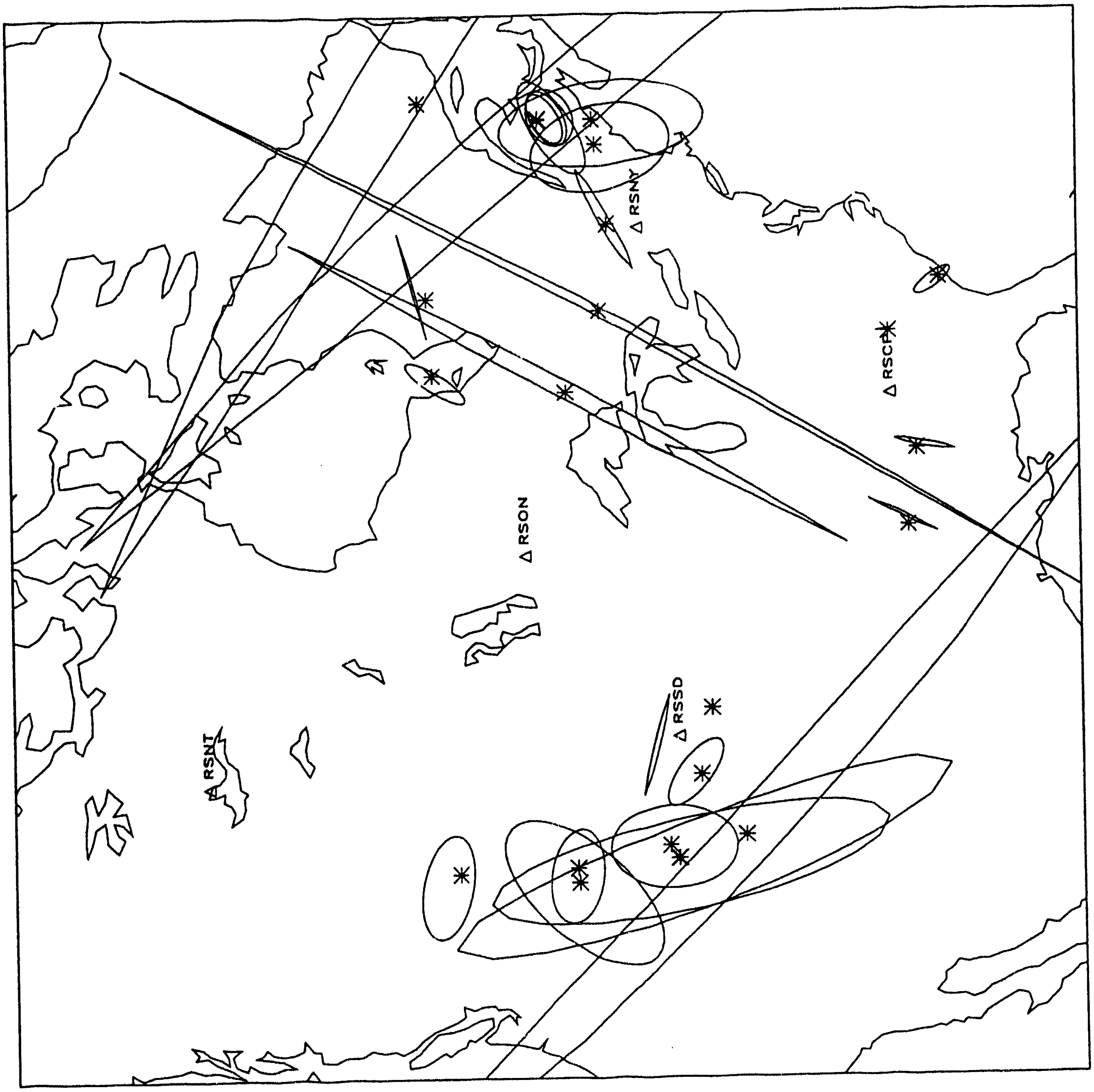


Figure 12. Location error ellipses with 99% confidence and the actual location of the event as listed in Table 2, and for events located with two stations. Twenty-one of the 26 ellipses include the actual location of the event.

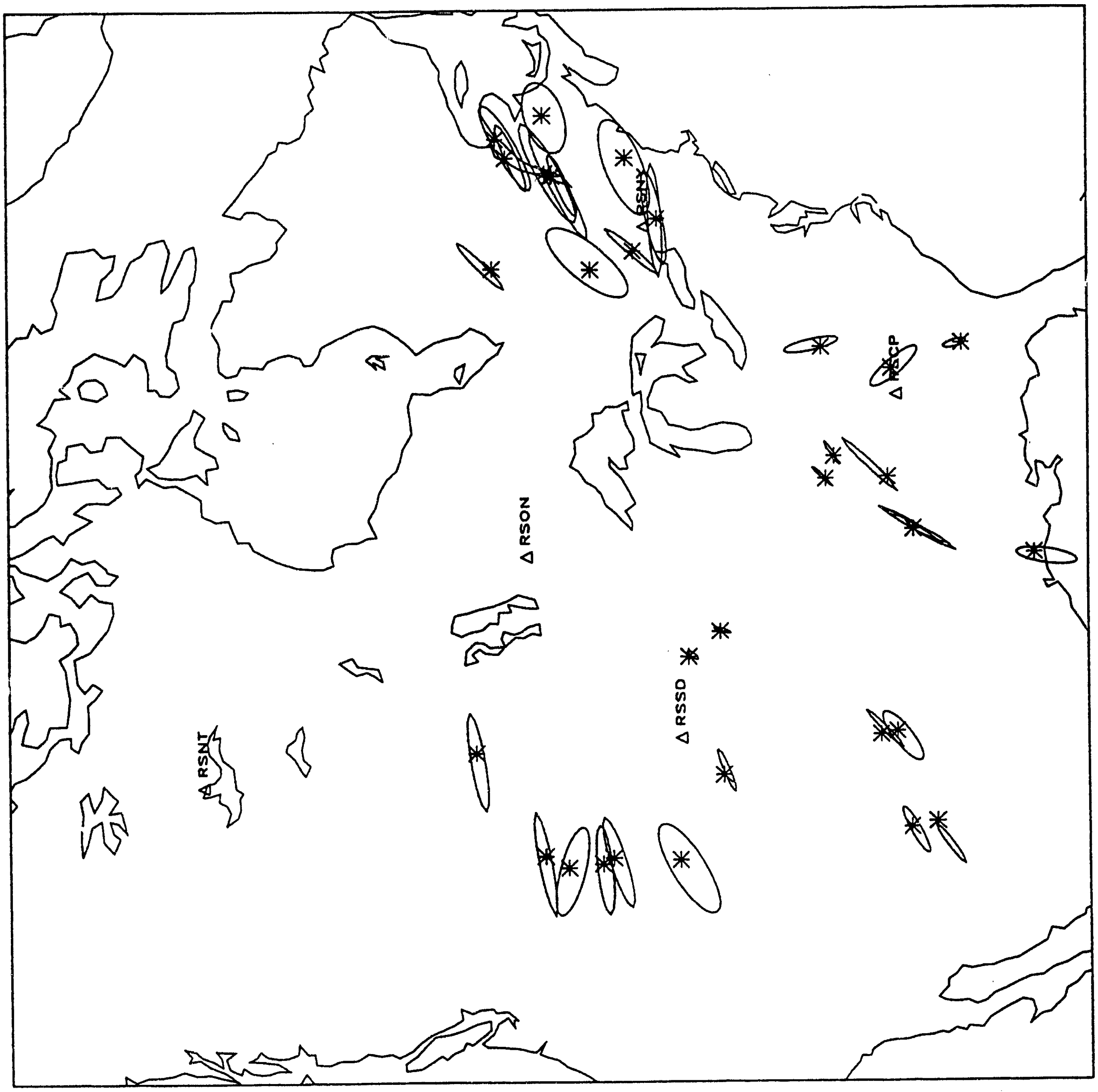


Figure 13. Location error ellipses with 99% confidence and the actual location of the event as listed in Table 2, and for events located with three or more stations. Thirty of the 33 ellipses include the actual location of the event.

SINGLE STATION DETECTION AS A FUNCTION OF MAGNITUDE AND DISTANCE

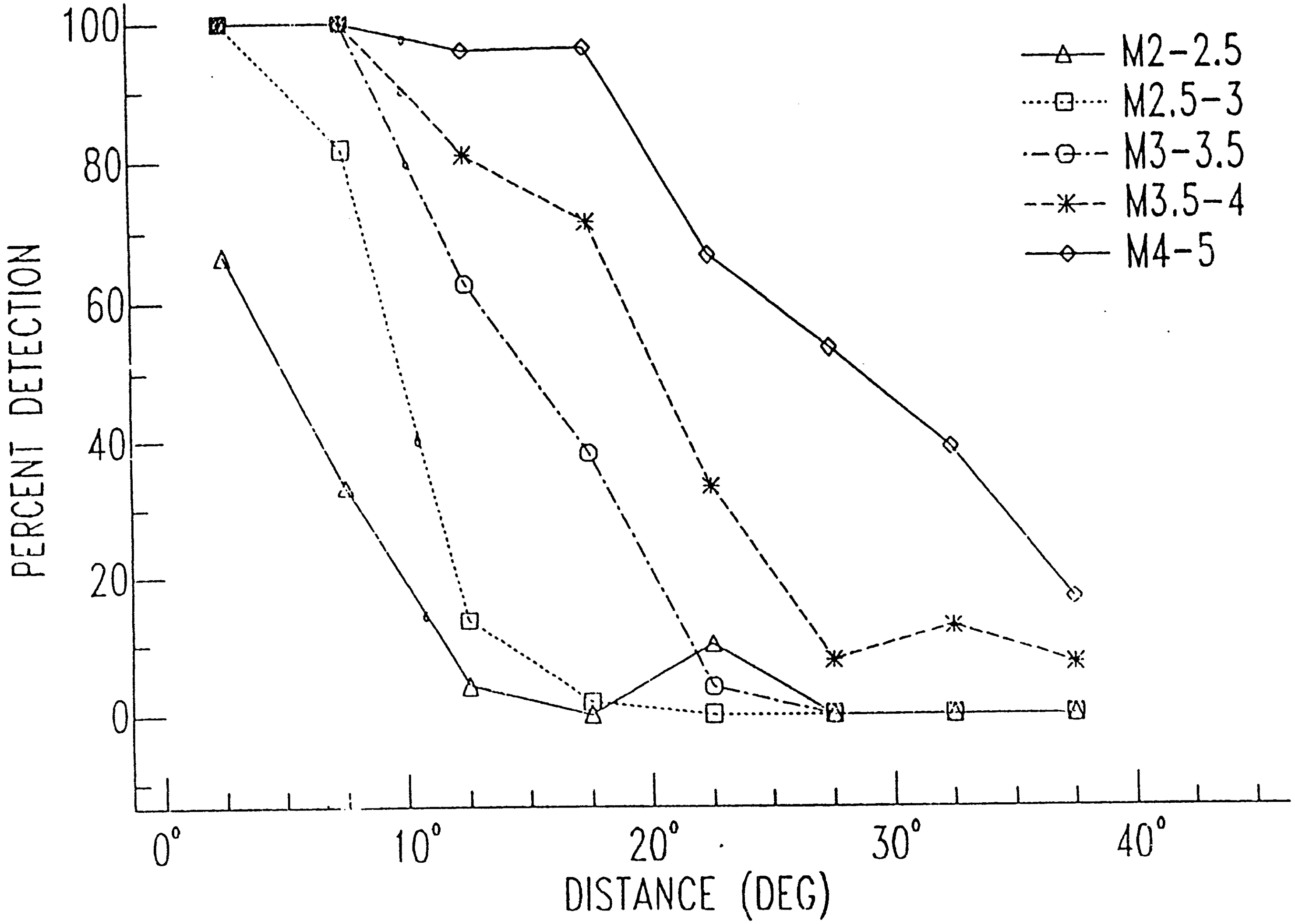


Figure 14. Shows the percentage of phases (of 848 possible) detected versus magnitude and recording distance (from Jarpe et. al., 1986).

COMPARISON OF DETECTION vs. DISTANCE FOR THE FIVE STATIONS

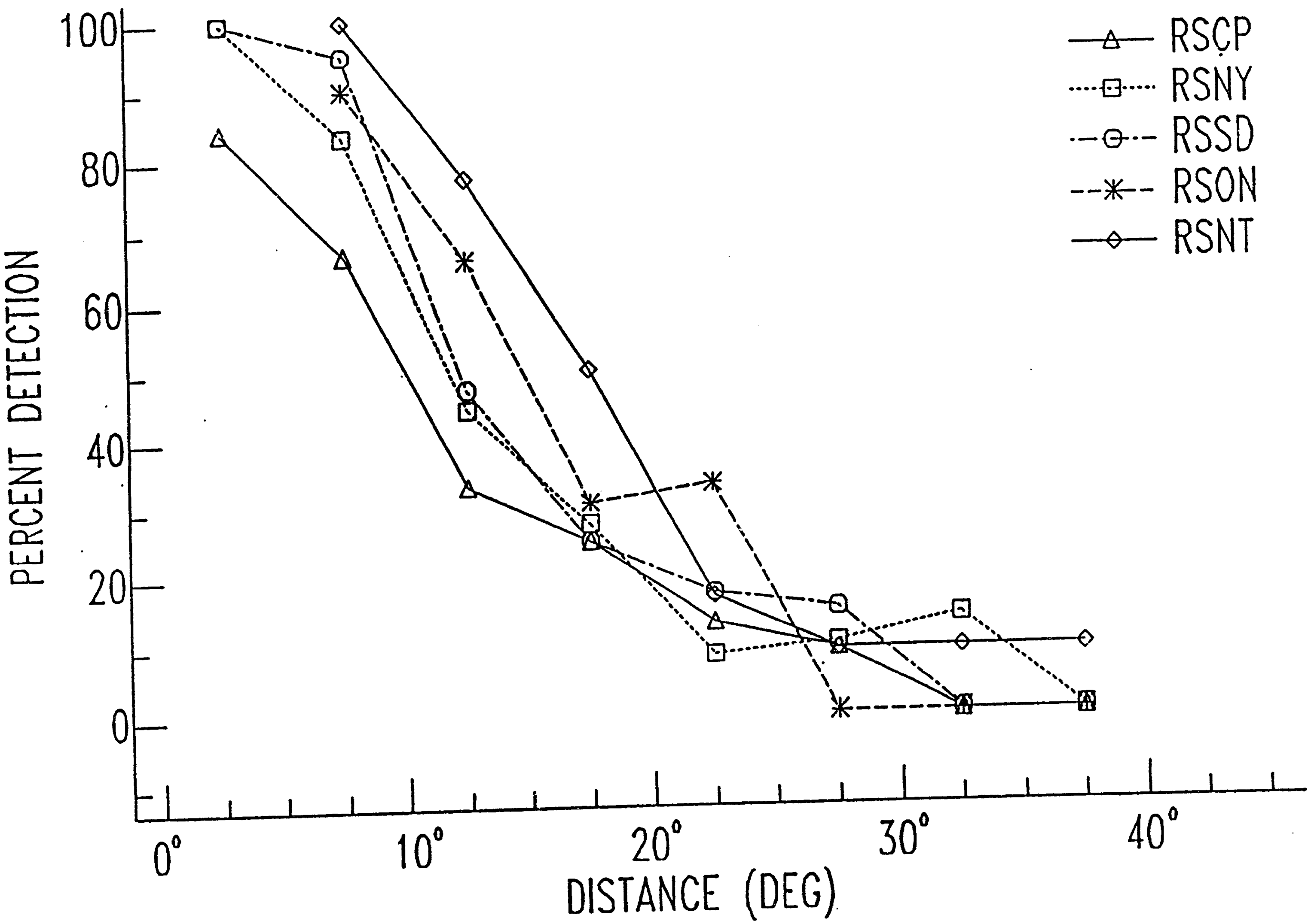


Figure 15. Shows the percentage of detections versus recording distance, irrespective of magnitude, for each station (from Jarpe et al.,1986).

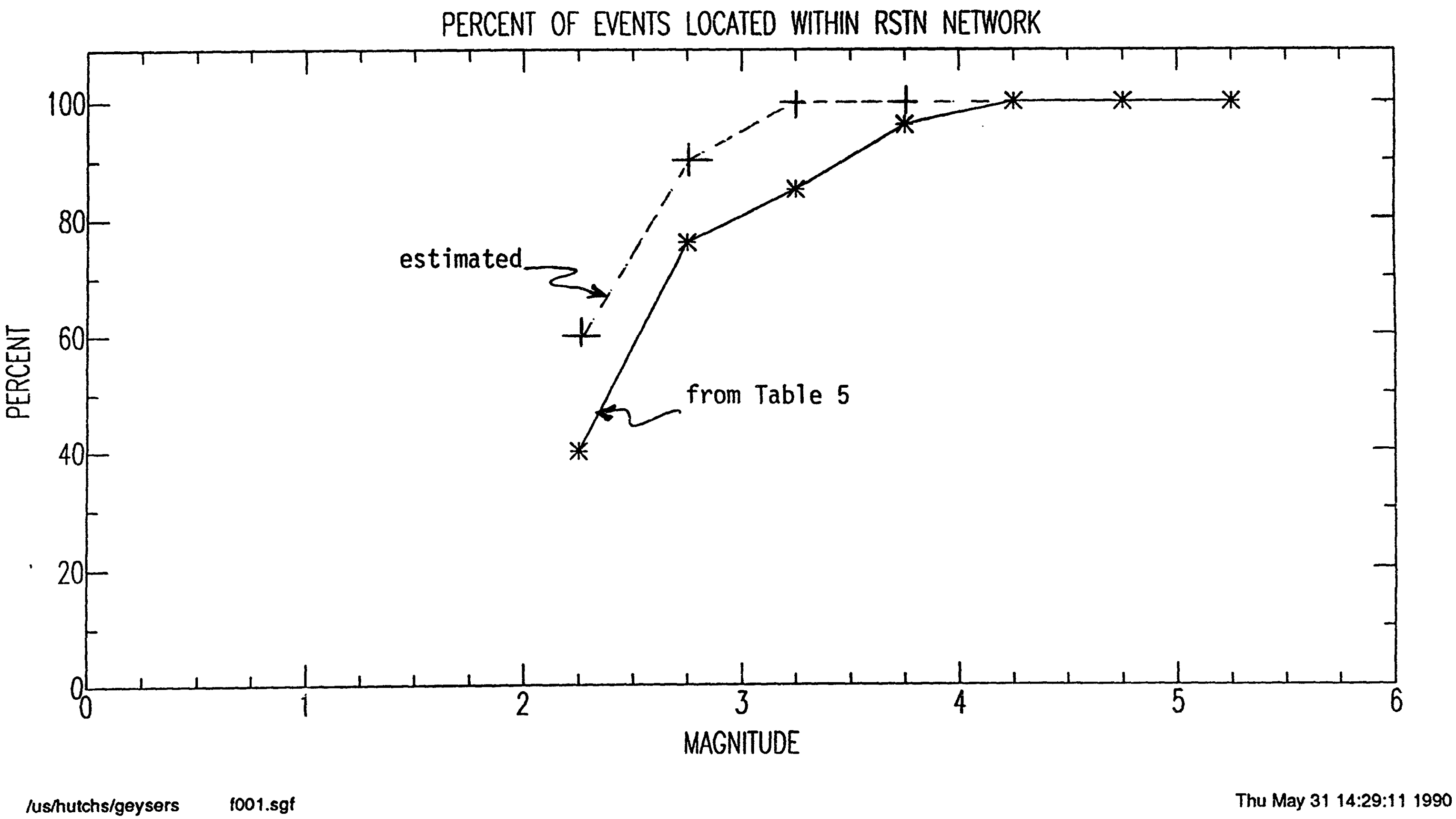


Figure 16. Estimate percentage of events that can be located as a functionof magnitude, and percentage located in this study

Appendix A

Phase Arrival Times and Backazimuths

```
rscp pn 3 82268231829.94    0.0
rscp pg 1 82268231846.30    0.0
rscp sn 0 82268231928.90    0.0
rscp lg 0 82268231953.10    0.0
rsny lg 2 82268232541.20    0.0
rson pn 2 82268232042.50    0.0
rson sn 2 82268232326.90    0.0
rson lg 1 82268232519.40    0.0
rssd pn 1 82268232003.40    0.0
rssd sn 1 82268232222.70    0.0
rssd lg 1 82268232350.90    0.0

rscp pn 2 82325163646.52    0.0
rscp pg 1 82325163701.13    0.0
rscp sn 0 82325163752.90    0.0
rscp lg 0 82325163816.30    0.0
rson sn 3 82325164148.00    0.0
rson lg 3 82325164337.20    0.0
rssd pn 3 82325163826.30    0.0
rssd sn 3 82325164043.20    0.0
rssd lg 1 82325164210.30    0.0

rscp pn 0 83019023202.25    0.0
rscp pg 0 83019023221.30    0.0
rscp sn 1 83019023303.80    0.0
rscp lg 0 83019023326.90    0.0
rsny lg 3 83019023909.20    0.0
rsnt lg 4 83019023909.20    0.0
rson pn 3 83019023416.40    0.0
rson sn 1 83019023700.70    0.0
rson lg 2 83019023847.30    0.0
rssd pn 1 83019023342.70    0.0
rssd sn 1 83019023556.40    0.0
rssd lg 0 83019023725.20    0.0

rscp pg 2 83036130901.8   261.4
rscp lg 1 83036130934.1   226.4
rssd lg 3 83036131622.7   125.7

rscp pn 3 83054085220.25    0.0
rscp pg 0 83054085225.75    0.0
rscp sn 0 830540853 0.59    0.0
rscp lg 0 83054085312.90    0.0
rsny lg 2 83054085820.80    0.0
rson pn 2 83054085456.20    0.0
rson pg 2 83054085619.60    0.0
rson sn 2 83054085732.70    0.0
rson lg 3 830540859 6.90    0.0
rssd sn 2 830540857 4.40    0.0
rssd lg 1 83054085837.20    0.0

rscp pn 0 83135051730.35    0.0
rscp pg 0 83135051741.69    0.0
rscp sn 0 83135051821.20    0.0
rscp lg 0 83135051839.70    0.0
rsny pn 0 83135051919.30    0.0
rsny sn 0 83135052131.20    0.0
rsny lg 0 83135052246.20    0.0
rson pn 0 83135051914.80    0.0
rson sn 0 83135052123.10    0.0
rson lg 0 83135052245.10    0.0
rssd pn 0 83135051911.70    0.0
rssd sn 0 83135052118.80    0.0
rssd lg 0 83135052237.00    0.0
```

```
rscp sn 2 8315422
rscp lg 2 8315422
rsny lg 4 8315422

rscp pg 3 83172135944.2
rscp sn 3 83172140025.8   278.5
rscp lg 2 83172140047.6   280.1
rssd lg 3 83172140454.7

rscp sn 3 83173040547.7   248.0
rscp lg 1 83173040558.0   224.0
rssd lg 4 83173040940.0

rsny sn 3 83343055046.20    0.0

rscp pg 0 84108044546.09    0.0
rscp sn 0 84108044621.83    0.0
rscp lg 0 84108044629.60    0.0
rsny lg 3 84108045053.10    0.0
rson pn 1 84108044743.50    0.0
rson sn 1 84108044959.60    0.0
rson lg 3 84108045125.80    0.0
rssd pn 3 84108044744.60    0.0
rssd sn 2 84108045006.00    0.0
rssd lg 1 84108045130.60    0.0
```

```
rsnt pn 3 82296055150.10    0.0
rsnt sn 3 82296055338.60  356.0

rsnt pn 0 82360081413.2   064.0
rsnt pg 4 82360081427.8   049.0
rsnt sn 0 82360081557.1   051.0

rsnt pn 3 82360084932.0
rsnt sn 2 82360085129.2   144.0
rsnt lg 3 82360085247.2

rssd lg 4 830821147
```

```
rssd pn 2 82296123026.50    0.0
rssd pg 1 82296123050.10  332.0
rssd sn 2 82296123138.40    0.0
rssd lg 0 82296123212.00  276.0
rson pn 3 82296123147.00    0.0
rson sn 2 82296123407.40    0.0
rson lg 2 82296123531.70  196.0

rsnt pn 4 82330200549.50    0.0
rsnt lg 2 82330201028.00    0.0
rson sn 4 82330200744.00    0.0
rson lg 2 82330200845.90    0.0
rssd pn 2 82330200333.61    0.0
rssd pg 1 82330200356.00    0.0
rssd sn 1 82330200438.70    0.0
rssd lg 0 82330200510.10    0.0

rsnt pn 1 83006023741.6   144.0
rsnt sn 1 83006023930.5   101.0
rsnt lg 2 83006024027.0   236.0
rson sn 0 83006023833.1   288.0
rson lg 0 83006023920.9   273.0
rssd pg 2 83006023733.5   389.0
rssd lg 2 83006023855.7   299.0

rsnt pn 2 83047062522.10    0.0
rsnt sn 1 83047062748.20    0.0
rsnt lg 1 83047062920.10    0.0
rson pn 1 83047062500.50    0.0
rson sn 0 83047062708.90    0.0
rson lg 0 83047062834.60    0.0
rssd pn 3 83047062356.30    0.0
rssd pg 3 83047062419.40    0.0
rssd lg 0 83047062552.00    0.0

rssd pn 3 83 47071532.54    0.0
rssd pg 0 83 47071553.80    0.0
rssd lg 0 83 470717 1.40    0.0
rson pn 2 83 470717 8.90    0.0
rson sn 2 83 47071942.40    0.0
rson lg 1 83 47072045.70    0.0
rsnt pn 2 83 470718 2.20    0.0
rsnt lg 2 83 47072245.80    0.0
rsnt sn 2 83 47072110.30    0.0

rscp pn 2 83 76073128.20    0.0
rscp lg 3 83 76073757.50    0.0
rson pn 0 83 76072857.90    0.0
rson pg 2 83 76072955.20    0.0
rson sn 0 83 76073118.80    0.0
rson lg 0 83 76073237.40    0.0
rssd pn 0 83 76072739.50    0.0
rssd pg 0 83 76072809.00    0.0
rssd lg 0 83 76072925.20    0.0

rson pn 3 83081045810.60    0.0
rson sn 3 83081050030.00  304.0
rson lg 3 83081050208.30  288.7
rssd lg 0 83081050034.60  280.2

rssd pg 3 83082014358.9   325.7
rssd lg 2 83082014526.9   278.8

rson pn 2 83313135611.80    0.0
rson lg 3 83313135956.60    0.0
rssd pn 0 83313135417.27  288.0
rssd pg 0 83313135425.03  276.0
rssd lg 0 83313135520.00  308.0

rscp pn 4 83354225708.90    0.0
rsnt pn 1 83354225651.50    0.0
rsnt lg 2 83354230223.00    0.0
rson pn 0 83354225538.70    0.0
rssd pn 0 83354225338.35    0.0
rssd pg 0 83354225352.82    0.0
rssd lg 0 83354225454.00    0.0

rson pn 2 833542325 9.20    0.0
rson lg 4 833542329 4.40    0.0
rssd pn 2 833542323 8.27  284.0
rssd pg 0 83354232321.48  276.0
rssd lg 0 83354232423.00  296.0

rson pg 3 83355202940.20    0.0
rssd pn 3 83355202711.40  324.0
rssd lg 2 833552029 7.20  316.0

rson pn 2 83356185918.40  196.0
rssd pn 1 83356185721.07  224.0
rssd pg 0 83356185733.11  256.0
rssd lg 0 83356185835.90  268.0

rssd pg 0 84084052319.80  256.0
rssd lg 1 84084052420.80  316.0
```

```
rscp pn 3 82287125659.90    0.0
rscp lg 1 82287130059.20    0.0
rsny lg 3 82287130547.30    0.0
rson pn 2 82287125730.10    0.0
rson sn 2 82287130020.00    0.0
rson lg 1 82287130204.80    0.0
rssd pn 1 82287125547.10    0.0
rssd pg 0 82287125619.50    0.0
rssd sn 0 82287125719.00    0.0
rssd lg 0 82287125809.50    0.0

rscp pn 0 82319030112.20    0.0
rscp sn 0 82319030323.10    0.0
rscp lg 0 82319030432.00    0.0
rsny pn 3 82319030219.10    0.0
rsny sn 2 82319030521.00    0.0
rsny lg 1 82319030708.60    0.0
rson pn 0 82319030020.40    0.0
rson sn 0 82319030149.90    0.0
rson lg 0 82319030236.60    0.0
rssd pn 0 82319025932.74    0.0
rssd pg 0 82319025943.32    0.0
rssd sn 0 82319030026.30    0.0
rssd lg 0 82319030044.20    0.0

rssd pn 2 82326101013.40  236.0
rssd pg 2 82326101044.40  232.0
rssd lg 1 82326101140.40  284.0

rscp pn 2 83044134844.90    0.0
rscp sn 2 83044135142.60    0.0
rscp lg 2 83044135334.50    0.0
rsnt pn 3 83044134926.10    0.0
rsnt sn 2 83044135332.40    0.0
rsnt lg 2 83044135539.40    0.0
rsny lg 3 83044135630.60    0.0
rson pn 0 83044134730.80    0.0
rson sn 0 83044134944.30    0.0
rson lg 0 83044135057.80    0.0
rssd pg 0 83044134526.55    0.0
rssd lg 0 83044134555.26    0.0

rscp pn 3 83 61232736.90    0.0
rscp sn 1 83 61233019.70    0.0
rscp lg 2 83 61233236.00    0.0
rsny lg 3 83 612337 9.70    0.0
rssd pg 0 83 61232626.40    0.0
rssd lg 0 83 61232840.90    0.0

rscp pn 0 83063063530.50    0.0
rscp sn 0 83063063754.90    0.0
rscp lg 0 83063063915.70    0.0
rsny pn 3 83063063623.70    0.0
rsny sn 2 83063063938.20    0.0
rsny lg 1 83063064121.30    0.0
rson pn 0 830630634 7.70    0.0
rson sn 0 83063063529.70    0.0
rson lg 0 83063063613.10    0.0
rssd pn 0 83063063310.47    0.0
rssd pg 0 83063063318.72    0.0
rssd lg 0 83063063357.74    0.0

rscp pn 3 83093045834.30    0.0
rscp lg 3 83093050233.80    0.0

rson sn 3 830930502 4.20    0.0
rson lg 3 830930504 5.90    0.0
rssd pn 2 83093045727.80    0.0
rssd pg 1 830930458 6.10    0.0
rssd sn 1 83093045856.10    0.0
rssd lg 0 830930500 0.20    0.0

rscp lg 3 83120074336.80    0.0
rson lg 3 83120074452.24   0.0
rssd pg 2 83120073742.60    0.0
rssd lg 2 631200740 5.20    0.0

rson pn 0 83126061711.2   239.7
rson sn 0 83126061854.8   229.5
rson lg 1 83126061955.0   224.2
rssd pg 1 83126061518.1   170.9
rssd lg 1 83126061535.9   170.6

rson lg 4 83226191640.70    0.0
rssd pn 1 832261910 5.26  212.0
rssd pg 0 83226191026.00  240.0
rssd sn 2 83226191130.50    0.0
rssd lg 0 83226191146.90  268.0

rson pn 2 83267170109.80    0.0
rson lg 3 83267170527.40  212.0
rssd pn 0 83267165900.75  272.0
rssd pg 0 83267165914.29    0.0
rssd sn 0 83267165956.60  244.0
rssd lg 0 83267170019.70  288.0

rson lg 3 83319123907.10    0.0
rssd pn 2 83319123344.00  240.0
rssd pg 4 83319123345.85  268.0
rssd lg 0 83319123409.00  320.0

rssd pg 2 84124182709.40  260.0
rssd lg 2 84124182820.40  244.0

rssd pg 2 84127020234.80  252.0
rssd lg 1 84127020343.90  246.0

rssd pg 3 84138091257.70  208.0
rssd lg 1 841380914 5.70  212.0
```

```
rsny pn 3 82276043353.40    22.0   corrected from 005.0
rsny sn 1 82276043555.90    86.0   correcter from 069.0
rsny lg 0 82276043713.10    66.0   corrected from 049.0
rson sn 4 82276043930.30    85.0
rson lg 4 82276044125.70    65.0

rsny pn 0 82299153301.21    69.9
rsny sn 1 82299153407.80     0.0
rsny lg 0 82299153441.10    53.5
rson lg 4 82299154052.80    75.5

rscp lg 2 82338161712.70     0.0
rsnt pn 3 82338161456.10     0.0
rsnt sn 2 82338162013.40     0.0
rsny pn 0 82338160934.61     0.0
rsny pg 0 82338160942.87     0.0
rsny sn 0 82338161022.50     0.0
rsny lg 0 82338161038.60     0.0
rson pn 3 82338161210.40     0.0
rson pg 1 82338161308.60     0.0
rson sn 1 82338161458.70     0.0
rson lg 0 82338161627.50     0.0
rssd lg 3 82338162058.90     0.0

rson pn 3 82351055107.8
rson sn 3 82351055354.4    078.0

rsny pg 2 82357070801.9    277.0   corrected from 260.0
rsny sn 1 82357070859.0    278.0   corrected from 261.0
rsny lg 0 82357070920.0    251.0   corrected from 234.0

rsny pn 1 83010213224.3    319.4   corrected from 302.4
rsny sn 1 83010213307.8
rsny lg 1 83010213321.4    308.4
rson sn 1 83010213542.7    094.7
rson lg 1 83010213644.2    100.0

rsny sn 3 83017082255.9    125.0   corrected from 108.0
rsny lg 3 83017082322.7    111.0   corrected from 094.0

rscp pn 3 83 19174013.40     0.0
rscp lg 2 83 19174538.20     0.0
rsnt pn 3 83 19174226.80     0.0
rsnt sn 3 83 19174734.20     0.0
rsny pn 0 83 19173729.60     0.0
rsny sn 0 83 19173841.30     0.0
rsny lg 0 83 19173916.20     0.0
rson pn 3 83 19173949.30     0.0
rson sn 1 83 19174257.70     0.0
rson lg 0 83 19174434.30     0.0

rsny pn 1 83020141840.7    054.0   corrected from 036.9
rsny sn 2 83020141939.3    051.0   corrected from 034.0
rsny lg 2 83020142008.2    057.0   corrected from 039.9

rsnt lg 3 83042160147.40     0.0
rsny pn 1 83042154825.14    50.0
rsny sn 0 83042154931.40     0.0
rsny lg 0 830421550 4.60     0.0
rson pn 3 83042155042.70     0.0
rson sn 3 83042155335.10     0.0
rson lg 2 83042155523.25     0.0

rsny pn 2 83133172730.28    67.7    corrected from 50.7
```

```
rsny pg 2 83133172810.60     0.0
rsny sn 2 83133172836.90    70.2    corrected from 53.2
rsny lg 2 83133172907.00    46.3
rson lg 2 83133173526.20    94.5

rscp pn 2 83133234508.40     0.0
rscp pg 1 83133234614.60     0.0
rscp lg 1 83133235011.90     0.0
rsnt pn 3 83133234745.50     0.0
rsnt sn 3 83133235325.30     0.0
rsny pn 0 83133234225.14     0.0
rsny sn 0 83133234332.20     0.0
rsny lg 0 83133234405.40     0.0
rson pn 2 83133234506.30     0.0
rson sn 3 83133234817.90     0.0
rson lg 3 83133235011.50     0.0
rssd pn 3 83133234657.60     0.0
rssd sn 2 83133235144.70     0.0
rssd lg 2 83133235432.00     0.0

rscp pn 3 83136020551.00
rscp pg 4 83136020723.90     0.0
rscp lg 2 83136021109.50
rsnt pn 4 83136020819.60     0.0
rsnt sn 3 83136021322.30
rsny pn 2 83136020304.30
rsny pg 3 83136020314.30
rsny sn 2 83136020355.20
rsny lg 0 83136020413.00
rson sn 2 83136020822.70
rson lg 0 83136021001.40
rssd sn 3 83136021155.00
rssd lg 2 83136021425.70

rscp pn 2 83149054923.90     0.0
rscp pg 4 83149054951.80     0.0
rscp sn 3 831490552 6.00     0.0
rscp lg 0 83149055323.30     0.0
rsnt pn 3 831490553 0.50     0.0
rsny pn 3 83149054636.87     0.0
rsny pg 0 83149054641.54     0.0
rsny sn 3 83149054705.36     0.0
rsny lg 0 83149054720.65     0.0
rson pn 4 83149054847.70     0.0
rson lg 0 83149055426.70     0.0
rssd pn 3 83149055131.40     0.0
rssd sn 3 83149055611.10     0.0
rssd lg 2 83149055822.40     0.0

rscp lg 3 83153063850.50     0.0
rsny pn 2 83153063126.30     0.0
rsny sn 3 83153063211.90     0.0
rsny lg 0 83153063224.40     0.0
rson sn 2 83153063651.70     0.0
rson lg 1 83153063822.20     0.0

rsny pn 1 83162134926.74     0.0
rsny sn 0 83162135033.20    37.0   corrected from 020.0
rsny lg 0 831621351 4.80    41.0   corrected from 024.0
rson lg 2 83162135723.50    92.0

rsny pn 0 83167163701.7    091.6   corrected from 074.6
rsny lg 0 83167163720.5    082.7   corrected from 065.7
```

```
rsny pn 2 83179080717.73    0.0
rsny sn 0 83179080822.30   45.0   corrected from 028.0
rsny lg 0 83179080856.70   77.0   corrected from 060.0
rson lg 2 83179081510.60   96.0

rsny pn 1 83224140959.43    0.0
rsny pg 0 83224141016.93   65.0   corrected from 048.0
rsny sn 2 83224141051.70   85.0   corrected from 068.0
rsny lg 0 83224141116.40   89.0   corrected from 072.0
rson pn 4 83224141319.40    0.0
rson sn 4 83224141323.30    0.0
rson lg 3 83224141811.70    0.0

rscp pn 0 83280102136.00    0.0
rscp pg 0 83280102215.00    0.0
rscp sn 0 83280102345.70    0.0
rscp lg 0 83280102455.50    0.0
rsny pg 0 83280101857.09    0.0
rsny pn 0 83280101859.20    0.0
rsny lg 0 83280101905.26    0.0
rsny sn 4 83280101907.76    0.0
rsnt pn 0 83280102534.00    0.0
rsnt sn 1 83280103033.10    0.0
rson pn 0 83280102209.40    0.0
rson sn 0 83280102447.80    0.0
rson lg 0 83280102617.60    0.0
rssd pn 0 83280102335.00    0.0
rssd sn 0 83280102736.90    0.0
rssd lg 0 83280102947.90    0.0

rscp sn 3 83280104439.60    0.0
rscp lg 3 83280104552.00    0.0
rsny lg 0 83280103957.40    0.0
rson sn 2 83280104603.30    0.0
rson lg 0 83280104714.20    0.0

rscp pn 2 83284041345.80    0.0
rscp sn 1 83284041555.00    0.0
rscp lg 0 83284041706.30    0.0
rsny pg 0 83284041114.27    0.0
rsny lg 0 83284041128.14    0.0
rson pn 2 83284041359.10    0.0
rson sn 0 83284041617.40    0.0
rson lg 0 83284041738.70    0.0
rssd sn 0 83284041905.60    0.0
rssd lg 0 83284042117.90    0.0

rsny pg 0 83305101713.19   28.0
rsny sn 0 83305101729.21   36.0
rsny lg 0 83305101732.34    0.0
rson lg 3 83305102405.40    0.0

rsny pn 1 83321153347.22   92.0
rsny sn 0 83321153453.00    0.0
rsny lg 0 83321153525.80   48.0
rson sn 3 83321154037.90    0.0
rson lg 1 83321154141.70   80.0

rsny pn 1 83338104933.02   96.0
rsny pg 0 83338104942.63  116.0
rsny sn 0 83338105014.30   68.0
rsny lg 0 83338105032.40   76.0
rson lg 3 83338105725.00   68.0
```

```
rsny pg 1 83355150457.64    0.0
rsny lg 1 833551505 8.30   82.0

rscp lg 3 83362123116.80    0.0
rsny pg 0 83362122511.73  332.0
rsny lg 0 83362122543.05  320.0
rssd pn 2 83362122848.70    0.0
```

```
rscp pg 0 82267221936.48    0.0
rscp lg 0 82267221950.83    0.0
rsny pn 3 82267222157.50    0.0
rsny sn 3 82267222405.60    0.0
rsny lg 3 82267222514.70    0.0
rssd lg 3 82267222820.00    0.0

rscp pg 0 83026140844.39    0.0
rscp lg 0 83026140925.80    0.0
rsny lg 3 83026141450.90    0.0
rssd lg 2 83026141758.80    0.0

rscp pg 1 83027221004.4   015.0
rscp lg 1 83027221025.9

rscp pg 1 83084024755.5   092.8
rscp lg 2 83084024829.3   143.6
rsny lg 2 83084025243.7   217.2  corrected from 200.2

rsny pn 1 83229140517.70    0.0
rsny sn 1 83229140651.10    0.0
rsny lg 1 83229140740.10    0.0
rson pn 2 83229140635.10    0.0
rson sn 2 83229140907.60    0.0
rson lg 1 83229141044.10    0.0
rssd sn 2 83229141015.70    0.0
rssd lg 2 83229141206.90    0.0

rscp pn 0 83289194253.50    0.0
rscp sn 0 83289194425.20    0.0
rscp lg 0 83289194524.90    0.0
rson pn 2 83289194531.90    0.0
rssd pn 1 83289194440.50    0.0
rssd sn 1 83289194734.40    0.0
rssd lg 2 83289194922.20    0.0

rscp pn 1 83310090337.30  140.0
rscp sn 3 83310090435.80    0.0
rscp lg 1 83310090457.80  116.0
rssd lg 4 83310091342.80    0.0
rsny lg 3 83310090842.80    0.0
```

Appendix B

Partial listing of REGLOC:

program to locate earthquakes with sparce data

```
c
      program regloc
c
c purpose: event location program utilizing both phase (one or more)
c          onsets and back-azimuth
c
c author: Lawrence Hutchings, LLNL
c
c          the program requires 2 input files:
c            "phases" and "stations"
c          the program produces 2 output files:
c            "summary" and "location"
c
c
      program regloc
c
      include 'regloc.cmn'
      data pn,sn,pg,sg,lg,bz/'pn','sn','pg','sg','lg','bz'/
      data blank/'    '/
c
c The file called "location" contains the output
c    it is written into from the main and 2 different subroutines:
c    "location" and "output"
c
      open(12, file='location')
c option to calculate travel time residuals from a fixed location. fxt = 1.0
c for fixed location, read location from file 'fxt.loc'.
      fxt = 0.0
      if(fxt.eq.1.0) then
      open(15, file='fxt.loc')
      read(15,*) ifyr,ifdy,ifhs,ifmin,fsec,flat,flon,fh
      endif
c ljh
c convention on latitude and longitude
c  latitue positive for north
c  longitude positive for east
c
c Number of location variables (lat,long,origin-time,depth) =  np
c      (for np=3, no iteration with depth; begins with 4,
c      is changed to 3)
      np=4
c
      pi2 = 6.28318530717958
      pi = 3.141592654
      dr = pi2/360.
c     constant to convert to ellipticity:
      gc=(1.-1./297.)**2
c
c ia = the order of g (g is a constant in inversion routine)
c  if np=4, ia=10
c  if np=3, ia=6
c
      ia=10
c
c
c To constrain depth to given value, idc = 1
c To make depth to be a free parameter, idc = 0
c idc is read from an input file
c For pure least squares, set damp = 0 (=1 otherwise)
c
c catlog: reads in a catalog of previously located events from
c          a file called "summary"
c
      call catlog
c
c velsta: reads in stations and velocity model info from a file
c          called "stations"
c
      call velsta
c
      if(idc.eq.0) go to 10
      np=3
      ia=6
c
c rearriv: reads in phase information
c          from the file "phases"
c
   10 irtn = 0
      call rearriv
c
      if (irtn .eq. 401) goto 401
      if (irtn .eq. 402) goto 402
c
c locate: does the location and writes to a file called "location"
c
      call locate
c
c rslmtx: Also writes into "location" , the covariance, data resolution
c         and model resolution matricies, and other information..
c
      call rslmtx
c
c
c output: Also writes into "location" the final solution and error ellipse
c         information
      call output
c
c sumfile: updates the file called "summary" to include the new location
c          a running catalog
c
      call sumfile
c
      go to 10
  401 write(12,1001)
      write(6,1001)
 1001 format('insufficient data to determine a solution')
      go to 10
  402 stop
      end
```

```
      subroutine rslmtx
      include 'regloc.cmn'
      real npl
c
c  g(ia)  constants of inversion matrix
c  ia    order of g
c  np    number of parameters solved for
c  nf    number of degrees of freedom: nf = iobs-np
c  idgt  elrments of inver correct to idgt decimal places.
c
c approach to calculating sigma for error ellipses:
c use BAZ for sigmad and sigmap if location is with 1 station
c use BAZ for sigmad only if location is with 2 stations
c don't use BAZ for either sigmad or sigmap if location is with 3 or more stations.

c estimate true variance, first prior variance
      sigmap = 0.0
      npl = 0.0
      do 101 i = 1,nosta
      do 201 j = 1,nopha
      if(pha(j).eq.bz.and.nosta.gt.1) go to 201
      if(awt(j).eq.0.0) go to 201
      if(stat(j).ne.sta(i)) go to 201
      if(pha(j).ne.pn) go to 202
      npl = npl + float(nrpn(i))
      sigmap = sigmap + rpn(i)
      go to 201
  202 if(pha(j).ne.pg) go to 203
      npl = npl + float(nrpg(i))
      sigmap = sigmap + rpg(i)
      go to 201
  203 if(pha(j).ne.sn) go to 204
      npl = npl + float(nrsn(i))
      sigmap = sigmap + rsn(i)
      go to 201
  204 if(pha(j).ne.sg) go to 205
      npl = npl + float(nrsg(i))
      sigmap = sigmap + rsg(i)
      go to 201
  205 if(pha(j).ne.lg) go to 206
      npl = npl + float(nrlg(i))
      sigmap = sigmap + nrlg(i)
      go to 201
  206 if(pha(j).ne.bz) write(12,110) pha(j)
  110 format('phase ',a2,' misidentified')
      aakm = acos(sin(rt(i))*sin(rphi)+cos(rt(i))*cos(rphi)
     &*cos(rlam-rl(i)))
      distt = aakm*111.12/dr
c remember wt = 0.1 for baz for non-algebraic solutions
c and baz distance weighted
      if(irtn.ne.404) then
      wt = 0.1*(200./distt)
      sigmap = sigmap + wt*(distt**2/vlg**2)*rbz(i)*dr**2
      npl = npl + wt*float(nrbz(i))
      endif
      if(irtn.eq.404) then
      sigmap = sigmap + 0.1*(distt**2/vlg**2)*rbz(i)*dr**2
      npl = npl + float(nrbz(i))
      endif
  201 continue
  101 continue
      sigmap = sigmap/(npl - float(np))
c
c now estimate true variance
c
      if(npl+df.lt.7.) then
      sigmap = 12.0
      npl = 4.0
      if(df.le.1) then
      df = 1.0
      sigmad = float(df)*3.0
      endif
      endif
      sigma = (npl-float(np))*sigmap/(npl+df) +
     &sigmad/(npl+df)
c
c compute resolution matrices;
c note that G matrix used here is not the G in Menke (1984, Geophysical
c data analysis: discrete inverse theory).  We actually use his G-g here,
c but since G-g is a symetric matrix, the same relations hold.
c
      do 10 i = 1,10
      c(i) = 0.0
      mg(i) = 0.0
      q(i) = 0.0
      v(i) = 0.0
      w(i) = 0.0
      d(i) = 0.0
   10 g(i) = g(i)*sigma
c
c     g*q-trans
c
      c(1) = g(1)**2  + g(2)**2 + g(4)**2 + g(7)**2
      c(2) = g(1)*g(2)+g(2)*g(3)+g(4)*g(5)+g(7)*g(8)
      c(3) = g(2)**2  + g(3)**2 + g(5)**2 + g(8)**2
      c(4) = g(1)*g(4)+g(2)*g(5)+g(4)*g(6)+g(7)*g(9)
      c(5) = g(2)*g(4)+g(3)*g(5)+g(5)*g(6)+g(8)*g(9)
      c(6) = g(4)**2  + g(5)**2 + g(6)**2 + g(9)**2
      if(ia.eq.6) go to 1
      c(7) = g(1)*g(7)+g(2)*g(8)+g(4)*g(9)+g(7)*g(10)
      c(8) = g(2)*g(7)+g(3)*g(8)+g(5)*g(9)+g(8)*g(10)
      c(9) = g(4)*g(7)+g(5)*g(8)+g(6)*g(9)+g(9)*g(10)
      c(10)= g(7)**2  +g(8)**2  +g(9)**2  +g(10)**2
    1 do 20 i = 1,ia
   20 ul(i) = c(i)
      call linvlp(ul,np,v,idgt,dd1,dd2,ier)
c
c covariance matrix
c sqrt(sigma) to compensate for normalization
c
      do 30 i = 1,ia
   30 q(i) = sigma*v(i)
      write(12,100)
     *q(1),q(2),q(4),q(7),
     *q(2),q(3),q(5),q(8),
     *q(4),q(5),q(6),q(9),
     *q(7),q(8),q(9),q(10)
  100 format(/,'covariance matrix'//,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4)
c model resolution matrix
c
      mg(1) = v(1)*c(1)+v(2)*c(2)+v(4)*c(4)+v(7)*c(7)
      mg(2) = v(1)*c(2)+v(2)*c(3)+v(4)*c(5)+v(7)*c(8)
      mg(3) = v(2)*c(2)+v(3)*c(3)+v(5)*c(5)+v(8)*c(8)
      mg(4) = v(1)*c(4)+v(2)*c(5)+v(4)*c(6)+v(7)*c(9)
```

```
      mg(5)  = v(2)*c(4)+v(3)*c(5)+v(5)*c(6)+v(8)*c(9)
      mg(6)  = v(4)*c(4)+v(5)*c(5)+v(6)*c(6)+v(9)*c(9)
      if(ia.eq.6) go to 2
      mg(7)  = v(1)*c(7)+v(2)*c(8)+v(4)*c(9)+v(7)*c(10)
      mg(8)  = v(2)*c(7)+v(3)*c(8)+v(5)*c(9)+v(8)*c(10)
      mg(9)  = v(4)*c(7)+v(5)*c(8)+v(6)*c(9)+v(9,*c(10)
      mg(10)= v(7)*c(7)+v(8)*c(8)+v(9)*c(9)+v(10)*c(10)
    2 write(12,102)
     *mg(1),mg(2),mg(4),mg(7),
     *mg(2),mg(3),mg(5),mg(8),
     *mg(4),mg(5),mg(6),mg(9),
     *mg(7),mg(8),mg(9),mg(10)
  102 format(/,'model resolution matrix'//,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4)
c data resolution matrix
      w(1)  = g(1)*v(1)+g(2)*v(2)+g(4)*v(4)+g(7)*v(7)
      w(2)  = g(1)*v(2)+g(2)*v(3)+g(4)*v(5)+g(7)*v(8)
      w(3)  = g(2)*v(2)+g(3)*v(3)+g(5)*v(5)+g(8)*v(8)
      w(4)  = g(1)*v(4)+g(2)*v(5)+g(4)*v(6)+g(7)*v(9)
      w(5)  = g(2)*v(4)+g(3)*v(5)+g(5)*v(6)+g(8)*v(9)
      w(6)  = g(4)*v(4)+g(5)*v(5)+g(6)*v(6)+g(9)*v(9)
      if(ia.eq.6) go to 3
      w(7)  = g(1)*v(7)+g(2)*v(8)+g(4)*v(9)+g(7)*v(10)
      w(8)  = g(2)*v(7)+g(3)*v(8)+g(5)*v(9)+g(8)*v(10)
      w(9)  = g(4)*v(7)+g(5)*v(8)+g(6)*v(9)+g(9)*v(10)
      w(10)= g(7)*v(7)+g(8)*v(8)+g(9)*v(9)+g(10)*v(10)
c
    3 d(1)  = w(1)*g(1)+w(2)*g(2)+w(4)*g(4)+w(7)*g(7)
      d(2)  = w(1)*g(2)+w(2)*g(3)+w(4)*g(5)+w(7)*g(8)
      d(3)  = w(2)*g(2)+w(3)*g(3)+w(5)*g(5)+w(8)*g(8)
      d(4)  = w(1)*g(4)+w(2)*g(5)+w(4)*g(6)+w(7)*g(9)
      d(5)  = w(2)*g(4)+w(3)*g(5)+w(5)*g(6)+w(8)*g(9)
      d(6)  = w(4)*g(4)+w(5)*g(5)+w(6)*g(6)+w(9)*g(9)
      if(ia.eq.6) go to 4
      d(7)  = w(1)*g(7)+w(2)*g(8)+w(4)*g(9)+w(7)*g(10)
      d(8)  = w(2)*g(7)+w(3)*g(8)+w(5)*g(9)+w(8)*g(10)
      d(9)  = w(4)*g(7)+w(5)*g(8)+w(6)*g(9)+w(9)*g(10)
      d(10)= w(7)*g(7)+w(8)*g(8)+w(9)*g(9)+w(10)*g(10)
    4 write(12,103)
     *d(1),d(2),d(4),d(7),
     *d(2),d(3),d(5),d(8),
     *d(4),d(5),d(6),d(9),
     *d(7),d(8),d(9),d(10)
  103 format(/,'data resolution matrix'//,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4///,
     *f10.4,3x,f10.4,3x,f10.4,3x,f10.4)
c 95% confidence limits, see Menke pg.29
c
      ccot = sqrt(-2.*q(1)*alog(0.05*sqrt(2.0*pi)*sqrt(q(1))))
      cphi = sqrt(-2.*q(3)*alog(0.05*sqrt(2.0*pi)*sqrt(q(3))))
      clam = sqrt(-2.*q(6)*alog(0.05*sqrt(2.0*pi)*sqrt(q(6))))
      czz = sqrt(-2.*q(10)*alog(0.05*sqrt(2.0*pi)*sqrt(q(10))))
c
c
c error ellipse (see Flinn (1965) Revs. Geophy. V 3, N 1)
c
c
      ul(1) = q(3)
      ul(2) = q(5)
      ul(3) = q(6)
      nn = 2
      call linv1p(ul,nn,d,idgt,dd1,dd2,ier)
c orientation of axis
c
      aaph = atan2(2.*d(2),d(1)-d(3))/2.
      if(aaph.lt.0.) aaph = aaph + pi2
c
c constants of ellipse, normalized for kilometers
c
      aa = d(1)*cos(aaph)**2 + 2.*d(2)*sin(aaph)*cos(aaph)
     &     + d(3)*sin(aaph)**2
c
      bb = d(1)*sin(aaph)**2 - 2.*d(2)*sin(aaph)*cos(aaph)
     &     + d(3)*cos(aaph)**2
c
c probility from F statistic:  see CRC math tables p.273
        npl = npl + df
c
c 90 percent confidence
c       if(npl.ge.3..and.npl.lt.4.)    ff=  5.46
c       if(npl.ge.4..and.npl.lt.5.)    ff = 4.32
c       if(npl.ge.5..and.npl.lt.6)     ff = 3.78
c       if(npl.ge.6..and.npl.lt.7)     ff = 3.46
c       if(npl.ge.7..and.npl.lt.8)     ff = 3.26
c       if(npl.ge.8..and.npl.lt.9)     ff = 3.11
c       if(npl.ge.9..and.npl.lt.10)    ff = 3.01
c       if(npl.ge.10..and.npl.le.15.)  ff = 2.92 - (npl-10)*0.037
c       if(npl.gt.15..and.npl.le.30.)  ff = 2.67 - (npl-16)*0.012
c       if(npl.gt.30..and.npl.le.60.)  ff = 2.49 - (npl-30)*0.0033
c       if(npl.gt.60..and.npl.le.120.) ff = 2.39 - (npl-60)*0.00067
c       if(npl.gt.120..and.npl.le.200.) ff = 2.35 - (npl-120)*0.0005
c       if(npl.gt.200.)                 ff = 2.30
c
c 95 percent confidence
      if(npl.ge.3..and.npl.lt.4)     ff=  9.55
      if(npl.ge.4..and.npl.lt.5)     ff = 6.94
      if(npl.ge.5..and.npl.lt.6)     ff = 5.79
      if(npl.ge.6..and.npl.lt.7)     ff = 5.14
      if(npl.ge.7..and.npl.lt.8)     ff = 4.74
      if(npl.ge.8..and.npl.lt.9)     ff = 4.46
      if(npl.ge.9..and.npl.lt.10)    ff = 4.26
      if(npl.ge.10..and.npl.le.15.)  ff = 4.10 - (npl-10)*0.07
      if(npl.gt.15..and.npl.le.30.)  ff = 3.63 - (npl-16)*0.0207
      if(npl.gt.30..and.npl.le.60.)  ff = 3.32 - (npl-30)*0.0057
      if(npl.gt.60..and.npl.le.120.) ff = 3.15 - (npl-60)*0.0013
      if(npl.gt.120..and.npl.le.200.) ff = 3.07 - (npl-120)*0.0009
      if(npl.gt.200.)                 ff = 3.00
c 97.5 percent confidence
c       if(npl.ge.3..and.npl.lt.4)     ff = 16.04
c       if(npl.ge.4..and.npl.lt.5)     ff = 10.65
c       if(npl.ge.5..and.npl.lt.6)     ff = 8.43
c       if(npl.ge.6..and.npl.lt.7)     ff = 7.26
c       if(npl.ge.7..and.npl.lt.8)     ff = 6.54
c       if(npl.ge.8..and.npl.lt.9)     ff = 6.06
c       if(npl.ge.9..and.npl.lt.10)    ff = 5.71
c       if(npl.ge.10..and.npl.le.15.)  ff = 5.46 - (npl-10)*0.115
c       if(npl.gt.15..and.npl.le.30.)  ff = 4.69 - (npl-16)*0.034
c       if(npl.gt.30..and.npl.le.60.)  ff = 4.18 - (npl-30)*0.0083
c       if(npl.gt.60..and.npl.le.120.) ff = 3.93 - (npl-60)*0.0022
c       if(npl.gt.120..and.npl.le.200.) ff = 3.80 - (npl-120)*0.0014
c       if(npl.gt.200.)                 ff = 3.69
c
c 99.0 percent confidence
```

```
c      if(npl.ge.3..and.npl.lt.4)      ff = 30.82
c      if(npl.ge.4..and.npl.lt.5)      ff = 18.00
c      if(npl.ge.5..and.npl.lt.6)      ff = 13.27
c      if(npl.ge.6..and.npl.lt.7)      ff = 10.92
c      if(npl.ge.7..and.npl.lt.8)      ff =  9.55
c      if(npl.ge.8..and.npl.lt.9)      ff =  8.65
c      if(npl.ge.9..and.npl.lt.10)     ff =  8.02
c      if(npl.ge.10..and.npl.le.15.)   ff =  7.56 - (npl-10)*0.2
c      if(npl.gt.15..and.npl.le.30.)   ff =  6.23 - (npl-16)*0.056
c      if(npl.gt.30..and.npl.le.60.)   ff =  5.39 - (npl-30)*0.0137
c      if(npl.gt.60..and.npl.le.120.)  ff =  4.98 - (npl-60)*0.0032
c      if(npl.gt.120..and.npl.le.200.) ff =  4.79 - (npl-120)*0.0023
c      if(npl.gt.200.)                 ff =  4.61
c
c equation for ellipse
c
c     c22 = aa*dphi**2 + bb*dlam**2
c
      c22 = ff*2.*sigma
c
c semiaxes of the ellipse in km, axis in degrees from north
c
      el1 = sqrt(c22**2/aa)*111.12
      el2 = sqrt(c22**2/bb)*111.12
      axe = aaph/dr
      return
      end
```

```
      subroutine algeb
c
      character*2 pha1,pha2,bstat*4
      include 'regloc.cmn'
      ak = 0.0
      oot = 0.0
      ddkm = 0.0
      depth = 10.0
      if(nopha.eq.2.and.nbz.eq.2) go to 1003
      if(depth.gt.z3) go to 1002
      bstat = stat(1)
    1 do 100 i = 1,nopha
      astat = stat(i)
      if(astat.ne.bstat) go to 103
   14 if(pha(i).eq.bz.or.iwt(i).eq.4.or.pha(i).eq.'  ') go to 100
      pha1 = pha(i)
      at1 = at(i)
      j = i
      do 20 k = 1,nopha
      if(stat(k).eq.astat.and.pha(k).eq.bz) go to 21
   20 continue
      go to 100
   21 bzz = baz(k)*dr
      if(baz(k).gt.180.) bzz = pi2 - bzz
      do 10 jj = 1,nosta
   10 if(astat.eq.sta(jj)) go to 11
   11 j = j + 1
      if(j.gt.nopha) go to 100
      if(pha(j).ne.bz.and.iwt(j).ne.4.and.stat(j).eq.astat) go to 13
      go to 11
   13 pha2 = pha(j)
      at2 = at(j)
c pn and sn
      if(pha1.eq.pn.and.pha2.eq.sn.or.pha1.eq.sn.and.
     &  pha2.eq.pn) then
      po = vp3/vs3
      if(depth.le.z2) dkm=f1(vs3,vp3)*(abs(at2-at1)-(po-1.)*
     &  f(2.*z2-depth,vp3,vp1)-(po-1.)*f(2.*(z3-z2),vp3,vp2))
      if(depth.gt.z2) dkm=f1(vs3,vp3)*(abs(at2-at1)-(po-1.)*
     &  f(z2,vp3,vp1)-(po-1.)*f(2.*z3-z2-depth,vp3,vp2))
      if(depth.le.z2) ttl = dkm/vp3 + f(2.*z2-depth,vp3,vp1)+
     &  f(2.*(z3-z2),vp3,vp2)
      if(depth.gt.z2) ttl = dkm/vp3 + f(z2,vp3,vp1)+
     &  f(2.*z3-z2-depth,vp3,vp2)
      if(pha1.eq.pn) ot = at1 - ttl
      if(pha2.eq.pn) ot = at2 - ttl
      endif
c pg and lg
      if(pha1.eq.pg.and.pha2.eq.lg.or.pha1.eq.lg.and.
     &  pha2.eq.pg) then
      dkm = f1(vlg,vp1)*abs(at2-at1)
      ttl = dkm/vp1
      if(pha1.eq.pg) ot = at1 - ttl
      if(pha2.eq.pg) ot = at2 - ttl
      endif
c pn and pg
      if(pha1.eq.pn.and.pha2.eq.pg.or.pha1.eq.pg.and.
     &  pha2.eq.pn) then
      if(depth.le.z2) dkm = f1(vp1,vp3)*(abs(at2-at1)+
     &  f(2.*z2-depth,vp3,vp1) + f(2.*(z3-z2),vp3,vp2))
      if(depth.gt.z2) dkm = f1(vp1,vp3)*(abs(at2-at1)+
     &  f(z2,vp3,vp1) + f(2.*z3-z2-depth,vp3,vp2))
      ttl = dkm/vp1
      if(pha1.eq.pg) ot = at1 - ttl
      if(pha2.eq.pg) ot = at2 - ttl
      endif
c pn and lg
      if(pha1.eq.pn.and.pha2.eq.lg.or.pha1.eq.lg.and.
     &  pha2.eq.pn) then
      if(depth.le.z2) dkm = f1(vlg,vp3)*(abs(at2-at1)+
     &  f(2.*z2-depth,vp3,vp1) + f(2.*(z3-z2),vp3,vp2))
      if(depth.gt.z2) dkm = f1(vlg,vp3)*(abs(at2-at1)+
     &  f(z2,vp3,vp1) - f(2.*z3-z2-depth,vp3,vp2))
      ttl = dkm/vlg
      if(pha1.eq.lg) ot = at1 - ttl
      if(pha2.eq.lg) ot = at2 - ttl
      endif
c sn and pg
      if(pha1.eq.pg.and.pha2.eq.sn.or.pha1.eq.sn.and.
     &  pha2.eq.pg) then
      po = vp3/vs3
      if(depth.le.z2) dkm = f1(vs3,vp1)*(abs(at2-at1)-
     &  po*f(2.*z2-depth,vp3,vp1) - po*f(2.*(z3-z2),vp3,vp2))
      if(depth.gt.z2) dkm = f1(vs3,vp1)*(abs(at2-at1)-
     &  po*f(z2,vp3,vp1) - po*f(2.*z3-z2-depth,vp3,vp2))
      ttl = dkm/vp1
      if(pha1.eq.pg) ot = at1 - ttl
      if(pha2.eq.pg) ot = at2 - ttl
      endif
c sn and lg
      if(pha1.eq.sn.and.pha2.eq.lg.or.pha1.eq.lg.and.
     &  pha2.eq.sn) then
      po = vp3/vs3
      if(depth.le.z2) dkm = f1(vlg,vs3)*(abs(at2-at1)+
     &  po*f(2.*z2-depth,vp3,vp1) + po*f(2.*(z3-z2),vp3,vp2))
      if(depth.gt.z2) dkm = f1(vlg,vs3)*(abs(at2-at1)+
     &  po*f(z2,vp3,vp1) + po*f(2.*z3-z2-depth,vp3,vp2))
      ttl = dkm/vlg
      if(pha1.eq.lg) ot = at1 - ttl
      if(pha2.eq.lg) ot = at2 - ttl
      endif
      ak = ak + 1.
      ddkm = ddkm + dkm
      oot = oot + ot
      go to 11
  101 bstat = stat(i+1)
      go to 100
  103 if(ak.ne.0.) go to 102
      bstat = astat
      go to 14
  100 continue
      if(ak.eq.0.0) return
  102 if(ns.ge.2) irtn = 400
      dkm = ddkm/ak
      ot = oot/ak
      delta = dkm*dr/111.12
      rphi=asin(cos(bzz)*sin(delta)*cos(rt(jj))+
     &  cos(delta)*sin(rt(jj)))
      ang1 = cos(delta)-sin(rphi)*sin(rt(jj))
      ang2 = cos(rphi)*cos(rt(jj))
      if(ang1.gt.ang2) ang1 = ang2
      rlam = acos(ang1/ang2)
      if(baz(k).ge.180.) rlam = rl(jj) - rlam
      if(baz(k).lt.180.) rlam = rl(jj) + rlam
      arlat=tan(rphi)
      arlat=atan(arlat/gc)
      dphi=arlat/dr
      dlam=rlam/dr
```

```
      ihs = ifix(ot/3600.)
      min = ifix((ot-float(ihs*3600))/60.)
      sec = ot-float(ihs*3600+min*60)
      if(ns.ge.2.) return
      hm=char(ihs/10+48)//char(ihs-(ihs/10)*10+48)//char(min/10+48)
     &  //Char(Min-(Min/10)*10+48)
      write(12,313)
      write(6,313)
 313  format('algebraic location with backazimuth from one station')
      if(fxt.eq.1.) then
      dphi = flat
      dlam = flon
      depth = fh
      ot = float(((ifhs*60)+ifmin)*60)+fsec
      iyr = ifyr
      idy = ifdy
      write(12,315)
 315  format('location fixed')
      endif
      do 110 i = 1,nopha
      if(pha(i).eq.bz.or.stat(i).ne.bstat) go to 110
      call ttray(i)
      that(i) = ot + thtt(i)
      rrs(i) = at(i) - that(i)
 110  continue
      go to 1001
 1002 write(12,314)
 314  format('earthquake is in mantle, cant get algebraic solution')
      go to 1001
c two station algebric solution for two backazimuths
 1003 write(12,312)
 312  format('algebraic location with backazimuth from two stations')
 1001 return
      end
```

```
c
c  calculates hrs mns sec of origin time
c
c  arrival times are created in the array "at" as secs from origin time.
c  earliest arrival time is called "ot" and all times are relative to "ot".
c
      subroutine arvtim
c
      include 'regloc.cmn'
c
      if(ot.gt.(24.*3600.)) go to 5
      if(ot.lt.0.0) go to 10
      go to 20
   10 idy = idy -1
      if(idy.lt.1) iyr = iyr - 1
      if(idy.lt.1) idy = 365
      do 30 i = 1,nopha
   30 at(i) = at(i) + 24.*60.*60.
      ot = ot + 24.*60.*60.
      go to 20
    5 idy = idy + 1
      if(idy.gt.365) iyr = iyr + i
      if(idy.gt.365) idy = 1
      do 25 i = 1,nopha
   25 at(i) = at(i) -24.*60.*60.
      ot = ot - 24.*60.*60.
   20 ihs = ifix(ot/3600.)
      min = ifix((ot-float(ihs*3600))/60.)
      sec = ot-float(ins*3600+min*60)
      return
      end
```

```
      program baz
c
c disbaz azimuth calculations with positive for west and north
c                                 negative for east and south
      character ans*3,ans1*3,ans2*1
      data ans1,ans2/'yes','y'/
      pi2 = 6.2831853071795B
      dr = pi2/360.0
      gc=(1.-1./297.)**2
      const=57.2957795
    1 print *,'** elipitical earth distance - azimuth calculation **'
      print *,' '
      print *,'longitude -179 < < 180 positive for west'
      print *,'latitude -90 < < 90  positive north'
      print *,'latitude and longitude of event'
      read(5,*) elat,elon
      print *,'latitude and longitude of station'
      read(5,*) slat,slon
c
      elat1=atan(gc*tan(elat/const))
      elon1=elon/const
      slat1=atan(gc*tan(slat/const))
      slon1=slon/const
      delo = elon1 - slon1
      delta=acos(sin(elat1)*sin(slat1)+cos(elat1)*cos(slat1)*cos(delo))
      baz11=(sin(elat1)-cos(delta)*sin(slat1))/(sin(delta)*cos(slat1))
      baz1=acos(baz11)
      if(elon1.gt.slon1.and.abs(delo).le.pi2/2.) baz1 = pi2 - baz1
      if(slon1.gt.elon1.and.abs(delo).gt.pi2/2.) baz1 = pi2 - baz1
      belo = slon1 - elon1
      baz22=sin(belo)*cos(elat1)/sin(delta)
      baz2=asin(baz22)
      if(slat1.le.elat1) go to 10
      if(belo.gt.-pi2.and.belo.le.-3.*pi2/4.) baz2 = pi2/2. - baz2
      if(belo.gt.-3.*pi2/4..and.belo.le.-pi2/2.) go to 10
      if(belo.gt.-pi2/2..and.belo.le.-pi2/4.) baz2 = pi2 + baz2
      if(belo.gt.-pi2/4..and.belo.le.0.) baz2 = pi2/2. - baz2
      if(belo.gt.0..and.belo.le.pi2/4.) baz2 = pi2/2. - baz2
      if(belo.gt.pi2/4..and.belo.le.pi2/2.) go to 10
      if(belo.gt.pi2/2..and.belo.le.3.*pi2/4.) baz2 = pi2 + baz2
      if(belo.gt.3.*pi2/4..and.belo.le.pi2) baz2 = pi2/2. - baz2
   10 if(baz2.lt.0.) baz2 = pi2 + baz2
      baz1=baz1/dr
      baz2=baz2/dr
      bcz=atan2(baz22,baz11)
      baz3=bcz/dr
      if(baz3.lt.0.) baz3 = baz3 + 360.
      aa=-sin(slon1-elon1)*sin(elat1)*cos(slat1)*cos(bcz)**2/
     &(sin(elat1) - sin(slat1)*cos(delta))
     &- sin(slon1-elon1)*cos(elat1)**2*cos(slat1)*cos(bcz)**2/
     &(sin(elat1) - sin(slat1)*cos(delta))**2
      bb=-cos(slon1-elon1)*cos(elat1)*cos(slat1)*cos(bcz)**2/
     &(sin(elat1)-sin(slat1)*cos(delta))
      delta=delta*111.12/dr
      write(6,101) delta,baz1,baz2,baz3,aa,bb
  101 format('distance = ',f8.1,' km  baz1 = ',f5.1,'   baz2 = ',f5.1,
     &'    baz3 = ',f5.1/,'partial w.r.t. phi = ',f5.1,'    partial w.r.t.
     & lambda = ',f5.1)
      goto 1
      return
      end
```

```
c
c This routine reads in a list of previously located earthquakes
c in order to update the list with a new location.
c
      subroutine catlog
c
c Subroutine to read a list of previously located events
c  purpose:- to add a new event to a previous list
c
      include 'regloc.cmn'
c
      open(13,file='summary')
      read(13,109,end=120)
  109 format(//)
      nsum = 0
  110 format(a9,1x,f5.2,2x,f7.3,2x,f8.3,3x,f5.2,2x,i2,2x,f6.2,1x,f6.1,4x,
     &f5.1,4x,f6.1)
      do 105 i = 1,1000
      read(13,110,end=120) summ(i),asec(i),adphi(i),adlam(i),
     &  adepth(i),nss(i),sigmaa(i),ell1(i),axel(i),ell2(i)
  105 nsum = nsum + 1
  120 close(13)
      return
      end
c
```

```
function f(a,b,c)
f=a*sqrt(b*b-c*c)/(b*c)
return
end
```

```
function f1(a,b)
f1=abs(1./(1./a-1./b))
return
end
```

```
c This routine writes the output of each iteration of
c the routine locate.
c
      subroutine iterout
c
      include 'regloc.cmn'
c
      yr=char(iyr/10+48)//char(iyr-(iyr/10)*10+48)
      idyy = idy/100
      idyyy= (idy-idyy*100)/10
      idyyyy = idy - idyy*100 - idyyy*10
      dy=char(idyy+48)//char(idyyy+48)//char(idyyyy+48)
      hm=char(ihs/10+48)//char(ihs-(ihs/10)*10+48)//char(min/10+48)
     *//char(min-(min/10)*10+48)
      write(12,302) yr,dy,hm,sec,dphi,dlam,depth
      write(6,303) yr,dy,hm,sec,dphi,dlam,depth,rms2
  302 format('ot= ',a2,1x,a3,1x,a4,':',f4.1,'  lat= ',f7.3,
     & '  lon = ',f8.3,'  h = ',f4.1)
  303 format('ot= ',a2,1x,a3,1x,a4,':',f4.1,'  lat= ',f7.3,
     & '  lon = ',f8.3,'  h = ',f4.1,3x,'rms error: ',f6.1)
      return
      end
```

```
c This routine is the main algorithm.
c It calculates the variables and constants for the least squares
c inversion routine. These constants are:
c       g(.), r(.)  --- constants in the l.s. matrix
c
c
      subroutine locate
c
      include 'regloc.cmn'
      real nrms,npl
c
      rms1 = 10000.0
      iter = 1
c calculate prior variance as total from all arrivals, to be used if
c an arrival does not have enough data
      call sigmp(sigmapl)
c     begin main iteration loop
      write(12,301)
  301 format(/,'*** iteration results ***')
  101 ot1 = ot
      ibp = 0
      if(irtn.ne.404.and.fxt.ne.1.0) call iterout
      do 10 k=1,10
      r(k)=0.
   10 g(k)=0.
c tiral latitude and longitude (dlat,dlam) in geographic degrees
c convert to geocentric radians
      rphi=dr*dphi
      rphi=gc*tan(rphi)
      rphi=atan(rphi)
      rlam=dr*dlam
c convert to geocentric degrees for units of inversion
      dphi1=rphi/dr
      anph = 0.
      do 20 j = 1,nopha
      if(pha(j).eq.bz) then
      aakm = acos(sin(rt(i))*sin(rphi)+cos(rt(i))*cos(rphi)
     &*cos(rlam-rl(i)))
      distt = aakm*111.12/dr
      awt(j) = 0.1*200.0/distt
      endif
      if(abs(rrs(j)).lt.5.) go to 20
      if(pha(j).ne.bz) then
      if(iter.gt.10.and.abs(rrs(j)).gt.3.*rms2)
     &  awt(j) = awt(j) - .25
      endif
      if(awt(j).lt.0.0) awt(j) = 0.0
   20 anph = anph + awt(j)
      if(anph.lt.(float(np))) then
      do 11 j = 1,nopha
   11 if(iwt(j).ne.4.) awt(j) = 1.0
      endif
      do 40 i=1,nosta
      rlat=rt(i)
      rlo=rl(i)
      u(i)=sin(rlat)*sin(rphi)+cos(rlat)*cos(rphi)*cos(rlam-rlo)
      delta=acos(u(i))
      dkm=delta*111.12/dr
      do 30 j = 1,nopha
      if(stat(j).ne.sta(i)) go to 30
      if(pha(j).eq.bz) then
      sin1 = -cos(rphi)*sin(rlo-rlam)/sin(delta)
      cos1 = (sin(rphi)-cos(delta)*sin(rlat))/
     &(sin(delta)*cos(rlat))
      bcz = atan2(sin1,cos1)
      bcz1 = bcz/dr
      if(bcz1.lt.0.0) bcz1 = bcz1 + 360.
      at(j)=ot
      endif
c     compute travel time and slowness for pn,sn
      if(pha(j).ne.bz) then
      call ttray(j)
c     convert dtdz (sec/km) to dt/dz (sec/deg) units
      dtdz(j)=dtdz(j)*111.12
c     convert ray parameter to sec/deg
      p(j)=p(j)*111.12
      that(j)=ot+thtt(j)
      endif
      wt = awt(j)
      if(iter.lt.5) wt = 1.0
      if(pha(j).ne.bz) then
      rrs(j)=at(j)-that(j)
      endif
      if(pha(j).eq.bz) then
      rrs(j) = baz(j)-bcz1
      endif
      if(awt(j).eq.0.0) go to 30
c
c prior variance
      sigmap = 0.0
      if(pha(j).eq.pn) npl = float(nrpn(i) - np)
      if(pha(j).eq.pn) sigmap = rpn(i)/npl
      if(pha(j).eq.pg) npl = float(nrpg(i) - np)
      if(pha(j).eq.pg) sigmap = rpg(i)/npl
      if(pha(j).eq.sn) npl = float(nrsn(i) - np)
      if(pha(j).eq.sn) sigmap = rsn(i)/npl
      if(pha(j).eq.sg) npl = float(nrsg(i) - np)
      if(pha(j).eq.sg) sigmap = rsg(i)/npl
      if(pha(j).eq.lg) npl = float(nrlg(i) - np)
      if(pha(j).eq.lg) sigmap = rlg(i)/npl
      if(pha(j).eq.bz) npl = float(nrbz(i))*awt(i) - float(np)
      if(pha(j).eq.bz) sigmap = awt(i)*rbz(i)/npl
c
c if np is less than 10, then sigma is used for normalizing equations
c first iteration estimate sigma
c hardwire sigmap for backazm with np < 10
      if(pha(j).eq.bz) then
      if(npl.lt.7.or.iter.eq.1) sigmap = 350.0
      endif
      if(pha(j).ne.bz.and.npl.lt.7) sigmap = sigmapl
      su=sqrt(1.-(cos(delta))**2)
      if(pha(j).ne.bz) then
      a(j)=-p(j)*(sin(rlat)*cos(rphi)-cos(rlat)*sin(rphi)*
     *cos(rlam-rlo))/su
      b(j)=p(j)*(cos(rlat)*cos(rphi)*sin(rlam-rlo))/su
      endif
      if(pha(j).eq.bz) then
      a(j)=-cos(rphi)/(sin(bcz)*cos(rlat)*sin(delta))
      b(j)=cos(rlo-rlam)*cos(rphi)/(cos(bcz)*sin(delta))
c partial derivitive units are radian/radian
      endif
      g(1)=g(1)+wt/sigmap
      g(2)=g(2)+a(j)*wt/sigmap
      g(3)=g(3)+a(j)**2*wt/sigmap
      g(4)=g(4)+b(j)*wt/sigmap
      g(5)=g(5)+a(j)*b(j)*wt/sigmap
      g(6)=g(6)+b(j)**2*wt/sigmap
      r(1)=r(1)+rrs(j)*wt/sigmap
```

```
      r(2)=r(2)+a(j)*rrs(j)*wt/sigmap
      r(3)=r(3)+b(j)*rrs(j)*wt/sigmap
      if(idc.eq.1) go to 30
      cz=dtdz(j)
      g(7)=g(7)+cz*wt/sigmap
      g(8)=g(8)+a(j)*cz*wt/sigmap
      g(9)=g(9)+b(j)*cz*wt/sigmap
      g(10)=g(10)+cz**2*wt/sigmap
      r(4)=r(4)+cz*rrs(j)*wt/sigmap
   30 continue
   40 continue
c     add damping factor to diagonal elements of g
      g(1)=g(1)+damp
      g(3)=g(3)+damp
      g(6)=g(6)+damp
      g(10)=g(10)+damp
c     compute weighted RMS error and degrees of freedom df
c     this is used to compute sigmad (data variance) in rslmtx.f
c ----LJH 7/20/92 change df to be unit values, because weighted variance
c     doesn't use sum of weights, o.k. for rms
      df = 0.0
      rms = 0.0
      nrms = 0.0
      do 60 i=1,nopha
      if(awt(i).eq.0.0) go to 60
      if(pha(i).eq.bz.and.nosta.lt.3) then
      aakm = acos(sin(rt(i))*sin(rphi)+cos(rt(i))*cos(rphi)
     &*cos(rlam-rl(i)))
      distt = aakm*111.12/dr
      rms   = rms + awt(i)*((distt/vlg)*rrs(i)*dr)**2
      nrms = nrms + awt(i)
      endif
      if(pha(i).ne.bz) then
      rms = rms + awt(i)*rrs(i)**2
      nrms = nrms + awt(i)
      endif
c      df = df + awt(i)
      df = df + 1.0
   60 continue
      sigmad = rms
      rms = sqrt(rms/nrms)
      if(irtn.eq.404) go to 350
      if(fxt.eq.1.0) go to 350
c     solve normal equations
      do 50 i=1,ia
   50 ul(i)=g(i)
      call linvlp(ul,np,ginv,idgt,d1,d2,ier)
      call luelmp(ul,r,np,x)
      rms2 = rms
      if(nbz.ne.0.and.rms2.gt.rms1.and.iter.gt.10) go to 1000
      rms1 = rms2
c     convert depth adjustment from degrees to km
      x(4)=x(4)*111.12
      if(idc.eq.0) write(12,7) (x(j),j=1,np),rms2
      if(idc.ne.0) write(12,8) (x(j),j=1,np),rms2
    7 format(12x,'dt=',f5.1,' dlat=',f6.3,'   dlon=',f7.3,'     dz=',
     &f4.1,'   rms error:',f6.1)
    8 format(12x,'dt=',f5.1,' dlat=',f6.3,'   dlon=',f7.3,'   dz=0.0   r
     &ms error:',f6.1)
      aa=abs(x(1))
      bb=abs(x(2))
      cc=abs(x(3))
      dd=abs(x(4))
      if(bb.lt..001.and.cc.lt..001.and.aa.lt.0.1.and.dd.lt.1.)
     &go to 1000
      dphil=dphil+x(2)
      if(dphil.le.90.) go to 3000
      dphil=180.-dphil
      go to 4000
 3000 if(dphil.ge.-90.) go to 2000
      dphil=-180.-dphil
 4000 if(dlam.gt.0.) dlam=dlam-180.
      if(dlam.le.0.) dlam=dlam+180.
 2000 dlam=dlam+x(3)
      if(dlam.gt.180.) dlam=-360.+dlam
      if(dlam.lt.-180.) dlam=360.+dlam
      ot=ot+x(1)
      call arvtim
      depth=depth+x(4)
c     prevent an airquake
      if(depth.lt.0.) depth=0.
      ibp = 1
      iter=iter+1
c conver back to geographic degrees
 1000 rphil=dphil*dr
      rphil=(tan(rphil))/gc
      rphil=atan(rphil)
      dphi=rphil/dr
      if(iter.gt.1000) write(12,307)
  307 format('******* too many iterations, cant find solution ********')
      if(iter.gt.1000) go to 350
      if(ibp.eq.1) go to 101
      if(rms2.lt.5..or.nst.lt.3) go to 350
      arrs = 0.0
      kwt = 0
      do 405 i = 1,nopha
      if(awt(i).eq.0.0.or.pha(i).eq.bz) go to 405
      kwt = kwt + 1
      if(abs(rrs(i)).lt.arrs) go to 405
      arrs = abs(rrs(i))
      jrr = i
  405 continue
      if(kwt.le.6) go to 350
      awt(jrr) = 0.0
      ibp = 1
      iter = 1
      write(12,313) stat(jrr),pha(jrr)
  313 format(/,'start iteration with phase ',a4,1x,
     & a2,' set to 4 wt')
      go to 101
  350 df = df-float(np)
      return
      end
```

```
      subroutine output
c
c This routine prints the final output and updates the
c  file summary
c
      include 'regloc.cmn'
      character alpha*9,bstat*4
c
      ahr = char(ihs/10+48)//char(ihs-(ihs/10)*10+48)
      mn = char(min/10+48)//char(min-(min/10)*10+48)
      summl = yr//dy//ahr//mn
      do 420 i = 1,nsum
      if(summ(i).ne.summl) go to 420
      summ(i) = summl
      amn(i) = mn
      asec(i) = sec
      adphi(i) = dphi
      adlam(i) = dlam
      adepth(i) = depth
      nss(i) = ns
      ell1(i) = ell
      ell2(i) = el2
      axel(i) = axe
      sigmaa(i) = sigma
      go to 421
  420 continue
      nsum = nsum + 1
      summ(nsum) = summl
      amn(i) = mn
      asec(i) = sec
      adphi(i) = dphi
      adlam(i) = dlam
      adepth(i) = depth
      nss(i) = ns
      ell1(i) = ell
      ell2(i) = el2
      axei(i) = axe
      sigmaa(i) = sigma
  301 format(/,'****** Final Solution ******')
  302 format(/,'stat pha wt  lwt   hrmn secs   dly   dis     res    baz
     & cbaz')
  303 format(a4,2x,a2,2x,i1,2x,f4.2,2x,a4,':',f4.1,1x,f5.1,2x,f6.1,
     &f6.1,9x,f4.0)
  304 format(a4,2x,a2,2x,i1,2x,f4.2,2x,a4,':',f4.1,1x,f5.1,2x,6x,f6.1)
  305 format(a4,2x,a2,2x,i1,2x,f4.2,14x,f6.1,13x,f4.0,1x,f4.0)
  306 format(a4,2x,a2,2x,i1,2x,f4.2,13x,21x,f4.0,2x,f4.0)
  307 format('95% confidence:'/,10x,'dot= :',f4.1,2x,'dlat=  ',f5.3,' dl
     &on=   ',f5.3,3x,'dh= ',f4.1,'variance:',f12.6)
  308 format('95% confidence:'/,10x,'dot= :',f4.1,2x,'dlat=  ',f5.3,' dl
     &on=   ',f5.3,3x,'variance: ',f6.2)
  309 format('95% confidence ellipse:'/,10x,f6.1,' km @',f5.1,'deg, and
     &',f6.1,' km @ +90 deg')
  310 format('ot= ',a2,1x,a3,1x, a4,':',f4.1,
     & '   lat= ',f7.3,'  lon= ',f8.3,'  h= ',f4.1)
  311 format('******** solution fixed ********')
c
  421 write(12,301)
      if(fxt.eq.1.0) write(12,311)
      write(12,310) yr,dy,hm,sec,dphi,dlam,depth
      if(ia.eq.10) write(12,307) ccot,cphi,clam,czz,sigma
      if(ia.eq.6) write(12,308) ccot,cphi,clam,sigma
      write(12,309) ell,axe,el2
      write(12,302)
      astat = ' '
      ddlta = 100000.0
      do 45 j = 1,nosta
      aakm = accs(sin(rt(j))*sin(rphi)+cos(rt(j))*cos(rphi)
     &*cos(rlam-rl(j)))
      dist(j) = aakm*111.12/dr
      if(dist(j).lt.ddlta) ddlta = dist(j)
      baz11=(sin(rphi)-cos(aakm)*sin(rt(j)))/(sin(aakm)*cos(rt(j)))
      baz22= sin(rlam-rl(j))*cos(rphi)/sin(aakm)
      bcz=atan2(baz22,baz11)/dr
      if(bcz.lt.0.0) bcz = bcz + 360.0
      do 40 i = 1,nopha
      if(stat(i).ne.sta(j)) go to 40
      if(pha(i).eq.bz) go to 41
c
c add back in travel time delay to put out arrival times
c
c find travel time delay for station and phase
      if(pha(i).eq.pn) tdly = tpn(j)
      if(pha(i).eq.sn) tdly = tsn(j)
      if(pha(i).eq.pg) tdly = tpg(j)
      if(pha(i).eq.sg) tdly = tsg(j)
      if(pha(i).eq.lg) tdly = tlg(j)
      at(i) = at(i) + tdly
c
      ihs=ifix(at(i)/3600.)
      min=ifix((at(i)-float(ihs*3600))/60.)
      hm=char(ihs/10+48)//char(ihs-(ihs/10)*10+48)//char(min/10+48)
     &//char(min-(min/10)*10+48)
      sec=at(i)-float(ihs*360 ,-float(min*60)
      if(stat(i).ne.astat) write(12,303) stat(i),pha(i),iwt(i),awt(i),
     &  hm,sec,tdly,dist(j),rrs(i),bcz
      if(stat(i).eq.astat) write(12,304) stat(i),pha(i),iwt(i),awt(i),
     &  hm,sec,tdly,rrs(i)
      go to 42
   41 if(stat(i).ne.astat) write(12,305) stat(i),pha(i),iwt(i),awt(i),
     &  dist(j),baz(i),bcz
      if(stat(i).eq.astat) write(12,306) stat(i),pha(i),iwt(i),awt(i),
     &  baz(i),bcz
   42 astat = stat(i)
   40 continue
   45 continue
      open(15,file='../figures/ellipses/e'//summl//'.error')
      write(15,101) summl,agm,ddlta,summl,summl,summl
  101 format('pole 50.0 -96.0'/,'window angles 21.0 21.0 21.0 21.0'/,
     &'title ','"EVENT ',a9,' mag ',f3.1,'  minimum distance',f7.1,
     &' km"'/,'id off'/,'sf 1 e',a9/,
     &'sa 1 n  3 size 0.7 1 f 3 size 0.8 loc r'/,'ellfiles e',a9/,
     &'bg sunw sgf'/,'map'/,'tty')
      close(15)
      open(15,file='../figures/ellipses/e'//summl//'.sta')
      bstat = '  '
      do 50 j = 1,nosta
      do 55 i = 1,nopha
      if(stat(i).ne.sta(j)) go to 55
      if(stat(i).eq.bstat) go to 55
      bstat = stat(i)
      slat =(tan(rt(j)))/gc
      slat = atan(slat)
      slat = slat/dr
      slon = rl(j)/dr
      write(15,102) sta(j),slat,slon
   55 continue
   50 continue
  102 format(a4,1x,f8.4,1x,f9.4)
```

96

```
      close(15)
      open(15,file='../figures/ellipses/e'//summl//'.ell')
      axel = axe - 90.
      if(el2.gt.el1) axel = axel + 90.0
      if(axel.gt.360.0) axel = axel - 360.0
      if(axel.lt.360.0) axel = axel + 360.0
      elp1 = amax1(el1,el2)
      elp2 = amin1(el1,el2)
      write(15,105) dphi,dlam,elp2,elp1,axel
 105  format(f8.4,1x,f9.4,1x,f6.1,1x,f6.1,1x,f5.1)
  53  close(15)
      close(16)
      return
      end
```

```
c This averages the back-az's from an individual station and
c creates the phase called "bz"
c
      subroutine phabaz
c
c location w:: ack-azimuth if phases only at 0, 1, or 2 stations
c
      include 'regloc.cmn'
c
      nopha1 = nopha
      nbz = 0
      do 10 i = 1, nosta
      avs = 0.0
      bzz = 0.0
      do 20 j = 1, jbz
      if(stbz(j).ne.sta(i)) go to 20
      avs = avs + 1.0
      bzz = bzz + bazz(j)
      if(avs.ne.1.0) go to 20
      nopha = nopha + 1
      pha(nopha) = 'bz'
      stat(nopha) = sta(i)
      awt(nopha) = 0.10
      iwt(nopha) = 3
      if(nst.ge.4.and.nopha1.ge.6) awt(nopha) = 0.0
      if(nst.ge.4.and.nopha1.ge.6) iwt(nopha) = 4
      nbz = nbz + 1
   20 continue
      if(bzz.ne.0.0) baz(nopha) = bzz/avs
   10 continue
      return
      end
```

```
c This routine reads arrivals, determines if sufficient info
c is available to locate and decides whether or not to use
c the back-azimuths estimates
c
      subroutine rearriv
c
      include 'regloc.cmn'
      character*2 bk
      data bk/'  '/
c
      open(11,file='phases')
      iter=1
c   read phases
      write(12,201)
      write(6,201)
      write(12,202)
  201 format(//,'***************** new event *****************'/)
  202 format('    phase information'//,'stat yr day hrmn sec  pha wt   d
     &ly    baz     bz-err')
  203 format(a4,1x,a2,1x,i1,1x,i2,i3,i2,i2,f5.2,2x,f5.1)
c first at relative to days/hrs of first arrival
c then in subroutine trial ot and at relative to days/00 hrs
c
c   nst: no. of stats with at least 1 at
c  nnst: no. of stats with at least 2 at
c   nbz: number of station with at least 1 baz
c    ns: number of stations in location
c nopha: total number of phases
c
      astat = ' '
      nopha = 0
      nst = 0
      nnst = 0
      nbz = 0
      ns = 0
      jbz = 0
      nn1 = 0
      nn2 = 0
      irtn = 0
      do 20 i = 1,100
      read(11,203,end=30) bstat,apha,jwt,iyr,idy,ihs,min,sec,bbaz
      if(bstat.eq.astat) go to 10
      if(nn1.ge.1) nst = nst + 1
      if(nn1.ge.2) nnst = nnst + 1
      if(nn2.ge.1) nbz = nbz + 1
      if(nn1.ge.1.or.nn2.ge.1) ns = ns + 1
      nn1 = 0
      nn2 = 0
      if(bstat.eq.blank) go to 30
   10 if(bbaz.ne.0.0) then
      nn2 = nn2 + 1
      jbz = jbz + 1
      bazz(jbz) = bbaz
      stbz(jbz) = bstat
       endif
      idyy = idy/100
      idyyy = (idy-idyy*100)/10
      idyyyy = idy - idyy*100 - idyyy*10
      yr=char(iyr/10+48)//char(iyr-(iyr/10)*10+48)
      dy=char(idyy+48)//char(idyyy+48)//char(idyyyy+48)
      hm=char(ihs/10+48)//char(ihs-(ihs/10)*10+48)//char(min/10+48)
     &//char(min-(min/10)*10+48)
      if(apha.eq.bk) go to 25
      if(jwt.ne.4) nn1 = nn1 + 1
      nopha = nopha + 1
      if(nopha.eq.1) then
      iyr1 = iyr
      idy1 = idy
      ihs1 = ihs
       endif
c find travel time delay for station and phase
      do 11 j = 1,100
   11 if(bstat.eq.sta(j)) go to 12
      print *,'station not on station list'
   12 if(apha.eq.pn) tdly = tpn(j)
      if(apha.eq.sn) tdly = tsn(j)
      if(apha.eq.pg) tdly = tpg(j)
      if(apha.eq.sg) tdly = tsg(j)
      if(apha.eq.lg) tdly = tlg(j)
c
      if(iyr.ne.iyr1) idy = idy1 + iyr-iyr1
      at(nopha) = float(((ihs*60)+min)*60)+sec
      at(nopha) = float((idy-idy1)*24*60*60) + at(nopha) - tdly
      iwt(nopha) = jwt
      if(jwt.eq.0) awt(nopha) = 1.0
      if(jwt.eq.1) awt(nopha) = 0.75
      if(jwt.eq.2) awt(nopha) = 0.50
      if(jwt.eq.3) awt(nopha) = 0.25
      if(jwt.eq.4) awt(nopha) = 0.0
      pha(nopha) = apha
      stat(nopha) = bstat
   25 if(bstat.ne.astat) go to 26
      if(bbaz.ne.0.0.and.apha.ne.bk) write(12,101) hm,sec,apha,jwt,
     &tdly,bbaz,tbz(j)
      if(bbaz.eq.0.0.and.apha.ne.bk) write(12,102) hm,sec,apha,jwt,
     &tdly
      if(bbaz.ne.0.0.and.apha.eq.bk) write(12,103) hm,bbaz,tbz(j)
      go to 27
   26 if(bbaz.ne.0.0.and.apha.ne.bk) write(12,104) bstat,yr,dy,hm,sec,
     & apha,jwt,tdly,bbaz,tbz(j)
      if(bbaz.eq.0.0.and.apha.ne.bk) write(12,105) bstat,yr,dy,hm,sec,
     & apha,jwt,tdly
      if(bbaz.ne.0.0.and.apha.eq.bk) write(12,106) bstat,yr,dy,hm,bbaz
  101 format(12x,a4,':',f4.1,2x,a2,2x,i1,1x,f5.1,3x,f5.1,4x,f5.1)
  102 format(12x,a4,':',f4.1,2x,a2,2x,i1,1x,f5.1)
  103 format(12x,a4,19x,f5.1,4x,f5.1)
  104 format(a4,1x,a2,1x,a3,1x,a4,':',f4.1,2x,a2,2x,i1,1x,f5.1,3x,f5.1
     &,4x,f5.1)
  105 format(a4,1x,a2,1x,a3,1x,a4,':',f4.1,2x,a2,2x,i1,1x,f5.1)
  106 format(a4,1x,a2,1x,a3,1x,a4,19x,f5.1,4x,f5.1)
   27 astat = bstat
      if(bbaz.ne.0.0) bazz(jbz) = bazz(jbz) - tbz(j)
   20 continue
   30 iyr = iyr1
      idy = idy1
      ihs = ihs1
      if(nopha.eq.0.and.nbz.eq.0) then
         irtn = 402
         return
      endif
      if(ns.eq.0) then
         irtn = 401
         return
       endif
      if(jbz.ne.0) call phabaz
      if(ns.ne.1) go to 31
      if(nst.eq.0.and.nnst.eq.0.and.nbz.eq.1) then
         irtn = 401
```

```
          return
       endif
       if(nst.eq.1.and.nnst.eq.0.and.nbz.eq.1) then
          irtn = 401
          return
       endif
       if(nst.eq.1.and.nnst.eq.1.and.nbz.eq.0) then
          irtn = 401
          return
       endif
       if(nst.eq.1.and.nnst.eq.0.and.nbz.eq.0) then
          irtn = 401
          return
       endif
c
         irtn = 404
       call algeb
         return
   31  if(ns.ne.2) go to 32
       if(nst.eq.2.and.nnst.eq.1.and.nbz.eq.0) then
          irtn = 401
          return
       endif
       if(nst.eq.2.and.nnst.eq.2.and.nbz.eq.0) then
          irtn = 401
          return
       endif
       if(nst.eq.2.and.nnst.eq.0.and.nbz.eq.0) then
            irtn = 401
            return
       endif
       if(nst.eq.1.and.nnst.eq.0.and.nbz.eq.1) then
           irtn = 401
           return
       endif
       if(nst.eq.1.and.nnst.eq.1.and.nbz.eq.1) then
           irtn = 401
           return
       endif
       if(nst.eq.0.and.nbz.eq.2) write(12,204)
  204  format('algebraic location for back-azimuth from two stations')
       if(nst.eq.0.and.nbz.eq.2) then
           irtn = 401
           return
       endif
   32  if(nst.eq.0) then
       ot = 0.0
       dphi = 45.0
       dlam = 90.0
       depth = 10.0
       return
       endif
       call trial
       return
       end
```

```
      subroutine sigmp(sigmapl)
      include 'regloc.cmn'
c estimate prior varianve for arrivals that do not have enough information
c
c  np    number of parameters solved for
c  nf    number of degrees of freedom: nf = iobs-np
c
      sigmapl = 0.0
      npp = 0
      do 101 i = 1,nosta
      do 201 j = 1,nopha
      if(awt(j).eq.0.0) go to 201
      if(stat(j).ne.sta(i)) go to 201
      if(pha(j).ne.pn) go to 202
      npp = npp + nrpn(i)
      sigmapl = sigmapl + rpn(i)
      go to 201
  202 if(pha(j).ne.pg) go to 203
      npp = npp + nrpg(i)
      sigmapl = sigmapl + rpg(i)
      go to 201
  203 if(pha(j).ne.sn) go to 204
      npp = npp + nrsn(i)
      sigmapl = sigmapl + rsn(i)
      go to 201
  204 if(pha(j).ne.sg) go to 205
      npp = npp + nrsg(i)
      sigmapl = sigmapl + rsg(i)
      go to 201
  205 if(pha(j).ne.lg) go to 206
      npp = npp + nrlg(i)
      sigmapl = sigmapl + nrlg(i)
      go to 201
  206 if(pha(j).ne.bz) write(12,110) pha(j)
  110 format('phase ',a2,' misidentified')
c do not include BAZ in prior variance
  201 continue
  101 continue
      sigmapl = sigmapl/float(npp - np)
      if(npp-np.lt.10) sigmapl = 1.0
      return
      end
```

```
c  writes the summary file
c----------------------------------------------------------------------
c
      subroutine sumfile
c
      include 'regloc.cmn'
c
      open(13,file='summary')
      write(13,332)
  332 format('                                   location summary'//
     & 'yrdayhrmn sec      lat        lon      depth stats varian
     &95 % error ellipse')
      do 403 i = 1,nsum
       if(sigmaa(i).gt.999.) sigmaa(i) = 999.
       if(ell1(i).gt.9999.) ell1(i) = 9999.9
       if(ell2(i).gt.9999.) ell2(i) = 9999.9
       if(i.ne.1) write(13,333) summ(i),asec(i),adphi(i),adlam(i),
     &  adepth(i),nss(i),sigmaa(i),ell1(i),axel(i),ell2(i)
       if(i.eq.1) write(13,334) summ(i),asec(i),adphi(i),adlam(i),
     &  adepth(i),nss(i),sigmaa(i),ell1(i),axel(i),ell2(i)
  403 continue
  333 format(a9,':',f5.2,2x,f7.3,2x,f8.3,3x,f5.2,2x,i2,2x,f6.2,1x,
     &f6.1,4x,f5.1,4x,f6.1)
  334 format(a9,':',f5.2,2x,f7.3,2x,f8.3,3x,f5.2,2x,i2,2x,f6.2,1x,
     &f6.1,'km @',f5.1,'deg ',f6.1,'km @+90deg')
      close(13)
      return
      end
```

```
      program table8
c
c creates tex file for table 8
c creatre summary and dist.err files first
      open(10,file='summary')
      open(11,file='dist.err')
      read(10,101) atiml
```

```
c funds a trial location
c
c-----------------------------------------------------------------
c
      subroutine trial
c
      include 'regloc.cmn'
c
      depth = 10.0
      call algeb
      if(irtn.eq.400.and.fxt.ne.1.0) return
      att = 1.0e+20
      do 10 i = 1,nopha
      if(pha(i).eq.bz) go to 10
      if(awt(i).eq.0.0) go to 10
      if(at(i).ge.att) go to 10
      att = at(i)
      ii = i
   10 continue
      do 20 j = 1,nosta
      if(sta(j).ne.stat(ii)) go to 20
      dphi = rt(j)/dr
      diam = rl(j)/dr
   20 continue
      if(fxt.eq.1.) then
      dphi = flat
      dlam = flon
      depth = fh
      att = float(((ifhs*60)+ifmin)*60)+fsec
      iyr = ifyr
      idy = ifdy
      endif
      if(att.ge.0.0) go to 30
      idy = idy -1
      ihs = 23 + ifix(att/3600.)
      if(idy.lt.1) iyr = iyr - 1
      if(idy.lt.1) idy = 365
      min = 24*60 + ifix(att/60) - ihs*60 - 1
      sec = 24.*60.*60.+att-float(ihs*3600)-float(min*60)
      go to 35
   30 ihs = ifix(att/3600.)
      min = ifix((att-float(ihs*3600))/60.)
      sec = att-float(ihs*3600+min*60)
      ot = float((ihs*60+min)*60)+sec
   35 do 40 i = 1,nopha
   40 at(i) = at(i) - att + ot
   41 return
      end
```

```
c determines the travel time partial derivatives
c-----------------------------------------------
c
      subroutine ttddel(j)
      include 'regloc.cmn'
c
      if(pha(j).ne.pn.and.pha(j).ne.sn) go to 50
      v1=vp1
      if(pha(j).eq.sn) v1=v1/po
      v2=vp2
      if(pha(j).eq.sn) v2=v2/po
      v3=vp3
      if(pha(j).eq.sn) v3=v3/po
c     find travel time branch
c     1st calculate 2 crossover distances, xc1,xc2
      xc1=(2.*z2-depth)*sqrt((v1+v2)/(v2-v1))
      tc1=xc1/v1
      ti1=f(2.*z2-depth,v2,v1)
      ti2=f(2.*z2-depth,v3,v1)+f(2.*(z3-z2),v3,v2)
      xc2=(ti2-ti1)*v2*v3/(v3-v2)
      tc2=tc1+(xc2-xc1)/v2
c      compute theoretical tts and partial derivatives
      dsc1=dkm-xc1
      dsc2=dkm-xc2
      if(dkm.gt.xc1 .and. dkm.le.xc2) go to 35
      if(dkm.gt.xc2) go to 45
      sxz=sqrt(dkm**2 + depth**2)
      thtt(j)=sxz/v1
      p(j)=dkm/(v1*sxz)
      dtdz(j)=depth/(v1*sxz)
      go to 40
   35 thtt(j)=dsc1/v2 + tc1
      p(j)=1./v2
      dtdz(j)=f1(v2,v1)
      go to 40
   45 thtt(j)=dsc2/v3 + tc2
      p(j)=1./v3
      dtdz(j)=f1(v3,v1)
      go to 40
   50 thtt(j) = dkm/vp1
      if(pha(j).eq.sg) thtt(j) = thtt(j)*po
      if(pha(j).eq.lg) thtt(j) = dkm/vlg
      if(pha(j).eq.bz) thtt(j) = dkm/vlg
      p(j) = 1.0/vp1
      if(pha(j).eq.sg) p(j) = p(j)*po
      if(pha(j).eq.lg) p(j) = 1.0/vlg
      if(pha(j).eq.bz) p(j) = 1.0/vlg
      dtdz(j) = 2.0*depth/(vp1*sqrt(dkm**2 + depth**2))
      if(pha(j).eq.sg) dtdz(j) = dtdz(j)*po
      if(pha(j).eq.lg) dtdz(j) = 0.0
      if(pha(j).eq.bz) dtdz(j) = 0.0
   40 return
      end
```

```
      subroutine ttray(j)
c
c     computes travel time, ray parameter, and dtdz
c     no raytracer, Pg computed as a direct arrival without depth
c
      include 'regloc.cmn'
      if(depth.gt.z3) go to 102
c pn and sn
      if(pha(j).eq.pn.or.pha(j).eq.sn) then
      if(depth.le.z2) thtt(j) = dkm/vp3 + f(2.*z2-depth,vp3,vp1)+
     &  f(2.*(z3-z2),vp3,vp2)
      if(depth.gt.z2) thtt(j) = dkm/vp3 + f(z2,vp3,vp1)+
     &  f(2.*z3-z2-depth,vp3,vp2)
      p(j) = 1.0/vp3
      if(depth.le.z2) dtdz(j) = f(-1.,vp3,vp1)
      if(depth.gt.z2) dtdz(j) = f(-1.,vp3,vp2)
      endif
      if(pha(j).eq.sn) then
      po = vp3/vs3
      thtt(j) = po*thtt(j)
      p(j) = po*p(j)
      dtdz(j) = po*dtdz(j)
      endif
c pg
      if(pha(j).eq.pg) then
      thtt(j) =dkm/vp1
      p(j) = 1.0/vp1
      dtdz(j) = 0.0
      endif
c lg
      if(pha(j).eq.lg) then
      thtt(j) = dkm/vlg
      p(j) = 1.0/vlg
      dtdz(j) = 0.0
      endif
c
      go to 101
  102 write(12,201)
  201 format('earthquake is in mantle, cant get calculate partials')
  101 return
      end
```

```
c reads in the velocity and station info
c----------------------------------------------------------------------
c
      subroutine velsta
c
c
c convention for sign of latitude and longitude:
c   latitude  + for north and  - for south
c   longitude + for east and - for west
      character alph*80
      include 'regloc.cmn'
      open(10,file='stations')
c input top of layer, velocity
      read(10,*) z1,z2,z3,vp1,vp2,vp3,vs1,vs2,vs3,vlg,idc,damp
      po = (vp1/vs1 + vp2/vs2 + vp3/vs3)/3.0
      po = 1.72
      write(12,101) z1,vp1,vs1,vlg,z2,vp2,vs2,z3,vp3,vs3,idc,po,damp
  101 format('  earthquake location'//, 'velocity model'/,'top of layer
     *  pvel   svel   lgvel'/,
     *3x,f4.1,8x,f3.1,4x,f3.1,5x,f3.1/,3x,f4.1,8x,
     *f3.1,4x,f3.1/,3x,f4.1,8x,f3.1,4x,f3.1//,
     *'idc=',i1,' average vp/vs ratio= ',f4.2,'  damping= ',f10.4)
c
      read(10,102) alph
  102 format(a120)
      write(12,103)
c
  103 format(//'    stations            prior 95% confidence errors(Km)
     &      travel time delays'/,' sta     lat       lon  erbaz  erpn er
     &sn erpg ersg erlg   tbz  tpn  tsn  tpg  tsg  tlg')
  104 format(a4,1x,f8.4,1x,f9.4,1x,f5.1,1x,f4.1,f5.1,f5.1,
     & f5.1,f5.1,2x,f5.1,f5.1,f5.1,f5.1,f5.1,f5.1)
c read in station locations, prior square residuals and sample number for
c phases recorded at each station.
c  erbaz: is sum of backazimuth errors squared, in degrees. erpn,
c   ect.: are the sum of travel time errors squared.
c
c read in travel time delays for each phase at each station.  tt delays
c are subtracted from traveltimes in rearriv.f
      nosta = 0
      do 10 i = 1,100
      read(10,*,end=11) sta(i),slat,slo,rbz(i),nrbz(i),rpn(i),nrpn(i),
     &rsn(i),nrsn(i),rpg(i),nrpg(i),rsg(i),nrsg(i),rlg(i),nrlg(i),
     &tbz(i),tpn(i),tsn(i),tpg(i),tsg(i),tlg(i)
      if(sta(i).eq.blank) go to 11
      nosta = nosta + 1
      rt(i)=dr*slat
      rt(i)=gc*tan(rt(i))
      rt(i)=atan(rt(i))
      rl(i)=dr*slo
c convert variance to 95% confidence error for gaussian distribution
c compute prior variance
c srbz computed for recording distance of 200 Km and Lg velocity
      srbz = (40000.0/vlg**2)*rbz(i)*dr**2
      srbz = srbz/(nrbz(i)-np)
      erbz = sqrt(-2.*srbz*alog(0.01*sqrt(2.0*pi)*sqrt(srbz)))
      srpn = rpn(i)/(nrpn(i)-np)
      erpn = sqrt(-2.*srpn*alog(0.05*sqrt(2.0*pi)*sqrt(srpn)))
      srsn = rsn(i)/(nrsn(i)-np)
      ersn = sqrt(-2.*srsn*alog(0.05*sqrt(2.0*pi)*sqrt(srsn)))
      srpg = rpg(i)/(nrpg(i)-np)
      erpg = sqrt(-2.*srpg*alog(0.05*sqrt(2.0*pi)*sqrt(srpg)))
      srsg = rsg(i)/(nrsg(i)-np)
      ersg = sqrt(-2.*srsg*alog(0.05*sqrt(2.0*pi)*sqrt(srsg)))
      srlg = rlg(i)/(nrlg(i)-np)
      erlg = sqrt(-2.*srlg*alog(0.05*sqrt(2.0*pi)*sqrt(srlg)))
c
      write(12,104) sta(i),slat,slo,erbz,erpn,ersn,erpg,ersg,erlg,
     &tbz(i),tpn(i),tsn(i),tpg(i),tsg(i),tlg(i)
   10 continue
   11 return
      end
```

DATE FILMED

4/7/94

END