HYPOCENTER 3.2

A Computer Program for Locating Earthquakes
Locally, Regionally and Globally

Barry R. Lienert

Hawaii Institute of Geophysics & Planetology
2525 Correa Rd Honolulu HI
Tel: (808) 956-7815
FAX: (808) 956-3188
E-mail: lienert@soest.hawaii.edu

October, 1994

CONTENTS

1 INTRODUCTION

HYPOCENTER is a Fortran program for locating local, regional and global earthquakes. The original version, described by *Lienert et al.*, **1986**, closely followed the format of HYPO71 (*Lee and Lahr*, **1978**) and was limited to locating local earthquakes ($\Delta < 1000$ km) due to the limitations of a rectangular coordinate system and "flat-earth" layered velocity model. In 1991, work commenced on adapting the program to locate global as well as regional and local earthquakes. The IASP91 software for calculating global travel times (*Kennett and Engdahl*, **1991**) had become available, making it possible to calculate the travel times for most of the global phases, rather than interpolating tables. The NEIC also released all of the ISC earthquake locations and their associated phase data for the period 1964-87 on a CD-ROM, allowing easy access to a data set of over 6 million arrivals using IBM PC-compatible computers. It became apparent that a global earthquake location program which would run on PC's as well as SUNs would be a valuable tool for comparing the ISC CD-ROM data with the IASP91 times, as well as those for new earth models.

A large number of new features have been added to HYPOCENTER in addition to the global location capability. These include a starting location algorithm, travel-time consistency tests, azimuthal data processing and a new input phase format as well as programs to translate additional input phase formats (ISC, HYPO71 and HYPOINVERSE). The program is also now capable of processing numbers of arrivals and stations limited only by the hard disk space available (the current default is 5000 stations and 2000 arrivals).

The program has been incorporated into the SEISAN database and processing system (*Havskov and Utheim*, **1992**) but is also distributed as a stand-alone package.

Notation Conventions: We have used capital letters throughout this manual for the names of programs and subroutines, as well as their associated files and variables. When describing the entry of an alphanumeric letter, such as the letter N, on a line, I enclose the letter in double quotes ("N"). This does not mean that the double quotes should be entered, only the value that they enclose. When a message occurs with output that is location-specific, I represent the number/tag with XXXX. Where names are case significant, as on UNIX machines, it will be specifically mentioned.

2 PROGRAM CAPABILITIES

- Simultaneous capability of locating local events, using a layered velocity model, or global events, using the IASP91 model. Layered model travel time calculations follow the procedure developed by *Eaton*, **1969** for a one-dimensional stack of layers having uniform thicknesses (no velocity inversions allowed). The following phases are recognized when using the layered model : P_g, P_n, P_b, S_g, S_n and S_b. Up to 40 different velocity models can be used for different events in a single file. Global travel times are calculated using the IASP91 model and software described by *Kennett and Engdahl*, **1991**. All the IASP91 phases (up to 100) are recognized. We have also included an optional auto-phase pick algorithm which matches all phases to the closest IASP91 phases after a specified number of iterations using first arrivals only. L_g and R_g are calculated using geocentric coordinates and therefore work both globally and locally.

- Solutions for global events and their errors are obtained using simultaneous damped-least-squares inversions of both primary P and PKP phases and secondary phases such as S, PP, PP etc.

- A consistency test and bisquare ($M1$) residual weighting to automatically identify and remove large residuals.

- A starting location algorithm which generates approximate locations of distant events using the azimuth and slowness of a plane wave, making it feasible to locate large global events using relatively small networks.

- Relative arrival time differences, e.g. for $S - P_n$, $S - P_g$, $L_g - P$, etc., may be optionally specified as input.

- Use of observed azimuths relative to each station and capability of one station location, if P and either S, L_g or R_g, and an azimuth are recorded.

- Calculation of magnitudes using either amplitude (M_L, and M_S),or coda (M_C).

- Some or all of the hypocentral parameters on each phase data header record may be used either to fix the solution or as a starting value.

- Station elevation corrections using either the layered model or the first layer of the IASP91 model.

- A statistical summary of residuals for all stations, mean r.m.s. residual for selected events and differences from previous locations.

- Output of the complete hypocentral covariance matrix for a user-specified confidence interval which allows plotting of hypocentral error ellipsoids.
- Less than a few seconds location time per event, when locating local events (<50 phases) with either a 486DX/66 or Sparc 10. For global events the time increases to approximately 0.5 seconds per fitted phase,

3 INSTALLATION

The hardware requirements are: PC: 386/486/Pentium with a math coprocessor and at least 4 Mb of total memory. SUN: No special requirements. For PC's, the software is distributed on two floppies, one with the files needed to run all the programs (executables) and essential input files, and the other disk containing the Fortran source code. If the SEISAN system is being used, HYPOCENTER is already included in it.

The Fortran sources on the PC and SUN are identical (see **8.6** for more details). For practical purposes, however, I have kept the distribution packages separate since the two IASP91 direct-access files are not compatible on the Sun and PC. The IASPEI Fortran source code to generate these latter two files is also included. More details are found in **8.5**.

The following files are included on distribution disk 1 (all are needed to run the HYPOCENTER executable, HYP.EXE).

HYP.EXE	executable file
IASP91.HED	IASP91 header file: read once at startup
IASP91.TBL	IASP91 random-access file: read every time a new depth is specified
STATION0.HYP	TEST parameters, station coordinates, velocity model file
EXAMPLE.DAT	Input phase file
DOSXMSF.EXE	Memory extender file (PC only)

The most straightforward way to access these files is to install them all in a single subdirectory (e.g., HYP). On a PC, the memory extender file must be in either this same directory, or one which is in the MS-DOS PATH command (in AUTOEXEC.BAT).

4 PROGRAM INPUT

4.1 The Parameter File, STATIONX.HYP

The input formats for the STATIONX.HYP file are similar to that of HYPO71 and HYPOINVERSE and are described in detail below. This file contains the following parameters:

1. TEST parameters, allowing changes in the programs default configuration
2. The station list, containing latitudes, longitudes, elevations and corrections for each station
3. The layered velocity model parameters
4. A default starting depth, which is used when the starting depth is not specified on the phase header record
5. The distance weighting parameters
6. The V_p/V_s ratio
7. The locating agency (blank if unused)

The parameter file must have the name STATIONX.HYP (upper-case on the SUN), where "X" is the character given in the header line (column 21) of the event phase set (see **4.2**). The default is set to "0" when this is blank, i.e., the default file when column 21 is blank is STATION0.HYP. This feature allows the velocity model (and TEST parameters) to be reset for different events while the program is running. Different STATIONX.HYP files, where "X" corresponds to the letters in column 21, must then be created. A sample STATION0.HYP file containing the ISC station and a commented set of TEST parameters is included with the program.
NOTE: When generating files containing different velocity models, they must contain an identical list of stations in order to get a correct list of average station residuals (in PRINT.OUT) at the end of a run.

4.1.1 Test Parameters

The TEST parameters are described below. For convenience. I have listed them in numerical order. They are all set to reasonable defaults and only 23 of them are normally needed in the parameter file (see **4.1.6** for this sample set). Note that some TEST parameter numbers are unused (e.g., 3-6). These are not used by HYPOCENTER, even though they may have significance to older programs such as HYPO71.

Columns	Format	Description
14-15	I2	Test parameter number
16-24	F9.4	Test parameter value
25-80		used for comments, e.g., parameter descriptions

The parameter list is terminated by a blank line. The TEST parameters all have default settings that will allow the program to run. However, their values are critical to the behavior of the program and I suggest including them with their default settings and comments, as in the example STATION0.HYP file, so that you are aware of the default values and can change them as needed.

TEST(2) Step length damping control. If the epicentral change (km) in a single iteration is larger than TEST(2), the damping will be increased until the change is < TEST(2). DON'T DECREASE THIS TO THE HYPO71 VALUE OF 50! It will slow the convergence process considerably and often have the opposite effect that you desire, destabilizing the iterative process.

default: TEST(2)=500.0

TEST(7) Coda magnitude (M_C) constant, a, where

$$M_C = a + b \log_{10} T + c\Delta + \Delta M \tag{1}$$

T is the coda length in seconds, Δ is the epicentral distance in km and ΔM is the station magnitude correction (see **4.1.2**)

default: TEST(7)=-0.87

TEST(8) Coda magnitude (M_C) constant, b.

If TEST(8) is negative, its positive value will be used for b and $\log_{10} T$ will be squared in the equation above for M_B. However, the individual stations magnitude values printed out during a run will still be calculated using the unsquared $\log_{10} T$.

default: TEST(8)=2.00

TEST(9) Coda magnitude (M_C) constant, c

TEST(9)=0.0035

TEST(11) Maximum no of iterations. The program should normally converge in 5-20 iterations, so the default value of 99 is ample.

default: TEST(11)=99.0

TEST(13) Increment in km at which auxiliary r.m.s. residual values are calculated. This provides a useful check that the minimum r.m.s. residual has been reached by perturbing the solution in the x, y and z directions by ± TEST(13) km and calculating the change in the r.m.s. residual. To disable calculation of these auxiliary values, which may take a significant time for global events, set TEST(13)=0.0

default: TEST(13)=20.0

TEST(30) Initial value of damping factor, xk. The damping factor is used to stabilize the solution when the partial derivatives become small. Increasing TEST(30) will slow the convergence of the solution, but also tends to prevent unstable oscillations during the initial iterations. It is the single most critical value affecting the behavior of the program. It effectively introduces a "trade-off" between the independence of the depth, latitude and longitude of the solution by decreasing their errors at the expense of correlating their values. Least-squares solutions, with no damping, can be obtained by setting TEST(30)=0.0. However, I do not recommend changing TEST(30), as I have generally found this value to be a good tradeoff between stability and speed of convergence.

default: TEST(30)=0.005

TEST(31) Maximum degrees of freedom -1 in the hypocentral solution. Set this to 2 for fixed depth solutions (to depth on phase headers), ,to -2 to fix all depths at value in STATION0.HYP, to 1 to fix all epicenters and depths (to values on phase headers) and to 0 to fix epicenters, depths and origin times (to values on phase headers). Do not use any other values of TEST(31).

default: TEST(31)=3.0 (nothing fixed)

TEST(32) Magnitude of parameter changes (km) below which convergence is assumed (increased by a factor of 10 initially).

default: TEST(32)=0.05

TEST(34) Minimum spread, s_{min} used to normalize residuals, dt, in the bisquare weighting. The spread, s, is defined (*Anderson*, **1982**) as

$$s = \max(A.\text{median}\{|dt|\}, s_{min}) \tag{2}$$

where $A = 1/0.6745$. The residual weights,w, are then calculated using

$$w = \left(1 - \left(dt'/c\right)^2\right)^2 \tag{3}$$

where $dt' = dt/s$ if median($|dt|$) $> s_{min}$ and $dt' = dt/s_{min}$ if median($|dt|$) $\leq s_{min}$ and c is a constant which determines the spread of the weighting. Since dt' is automatically normalized to the residual spread in a given data set, I do not recommend changing TEST(34) or TEST(35) without a thorough understanding of the statistics of your data. I also do not recommend using residual weighting for local events, as it significantly reduces the domain of convergence of the damped least-squares inversion.

default: TEST(34)=0.1

TEST(35) Bisquare weighting width, c used in the formula above.

default: TEST(35)=4.685

TEST(36) R.m.s. residual value below which bisquare residual weighting is applied to residuals for **local** events. For example, setting TEST(36)=5.0 will enable residual weighting for local events once the r.m.s. residual falls below 5 seconds. Use TEST(88) to similarly enable or disable the residual weighting for distant events.

default: TEST(36)=0.0 (disabled)

TEST(37) Maximum number of increases in the damping before fixing depth

default: TEST(37)=10.

TEST(38) = 0.0 for least squares errors, anything else, damped least squares errors with initial TEST(30) damping value

default: TEST(38)=0.0 (least squares errors)

TEST(39) Factor by which damping, xk, is increased when r.m.s. residual increases

default: TEST(39)=4.

TEST(40) Parameter that determines the depth origin of the coordinate system used for hypocentral depth as well as for the velocity model for **local** events. Internally, the program uses the maximum station elevation in STATIONX.HYP, MAXELV/1000 km, as the origin. If TEST(40)=0.0 (the default), the hypocentral depth referenced to sea level, which means that hypocenters are never allowed above sea level. MAXELV/1000 is then added (subtracted if negative) to the thickness of the top layer in the layered velocity model, i.e., the velocity above sea level is assumed to be the velocity in the model's top layer. MAXELV/1000 is also added to the hypocentral depth (subtracted if negative) when calculating travel times.
If TEST(40)=1, both the hypocentral depth and the velocity model are referenced to the maximum elevation station (this elevation may vary for different STATIONX.HYP files). Setting TEST(40)=1 is useful when different layers are needed for elevation corrections and also when events are above some of the stations, as may occur with Ocean Bottom Seismometer networks.

For distant events, the depth origin is always at sea level and setting TEST(40)=1 has no effect. The velocity in the top layer in the IASP91 model and its resulting vertical slowness is the used to calculate an elevation correction at each station which is added to the observed travel times.

default: TEST(40)=0.0 (origin at sea level)

TEST(41) Maximum distance (km) from nearest station at which hypocentral solutions will be generated. The iterations stop and an error message is printed if this limit is exceeded.

default: TEST(41)=20000.

TEST(43) Maximum r.m.s. residual an event may have for its residuals to be used in average station residual calculation. This doesn't affect the hypocenter calculation.

default: TEST(43)=15.0

TEST(44) R_g phase velocity in km/sec

default: TEST(44)=3.0

TEST(45) A location difference between the determined hypocenter and the hypocenter on the input phase header record is determined for each event. At the end of a run, the mean location differences and their standard deviations are calculated. The r.m.s residual of each event must be $<$ TEST(45) seconds for its location differences to be included in these means.

default: TEST(45)=50.0

TEST(46) If the number of non-zero weight phases for an event is $>$ TEST(46), the events location differences are included in the location difference means (see TEST(45) description).

default: TEST(46) = 3.0

TEST(50) Flag for using azimuth phases. Setting TEST(50)=0 disables the use of azimuths. However, it also disables the use of azimuths in the starting location, so it is often better to set the azimuth error, TEST(52), to a large value rather than setting TEST(50)=0.

default: TEST(50)=1.0 (use azimuths)

TEST(51) L_g phase velocity in km/sec

default: TEST(51)=3.5

TEST(52) Azimuthal error used in azimuth inversion (degrees). This parameter determines the relative weighting between azimuth residuals and time residuals. The default value of 10 means that an error of 10 degrees in azimuth will have the same relative effect on the solution as a one second error in an arrival time.

default: TEST(52)=10.0

TEST(53) Critical distance (km) which the starting location is moved out to if Moho phases (P_n, S_n) are specified.

default: TEST(53)=130.

TEST(56) A value of 1.0 enables starting location estimates from apparent velocity, distance, azimuths, etc. If TEST(56)=0.0 epicenter is taken 0.2 km NE of the first arrival station.

default: TEST(56)=1.0 (use starting locations)

TEST(57) If the geocentric distance to the event from any station exceeds TEST(57) km, the IASP91 routines are used to calculate travel times. However, a "D" or "L" indicator on the input phase header record will still force the use of the IASP91 or layered models, respectively.

default: TEST(57)=1500.

TEST(58) Maximum apparent velocity (km/sec) for phase data to be used. This feature was added to selectively disable *PKP* phases when using array data yielding apparent velocities.

default: TEST(58)=100.0

TEST(59) Critical distance (km) for *PKP* core phases. The starting location algorithm will use this, along with any available azimuth, to determining a start location when a non-zero weight *PKP* phase is specified.

default: TEST(59)=13000.

TEST(60) Seconds by which the arrival time difference between two adjacent stations can exceed the calculated travel time between them. These travel times are calculated using the layered (local) velocity model, even at teleseismic distances (see **8.4** for details). Setting TEST(60)=0 disables this initial consistency test.

default: TEST(60)=5.0 (enabled)

TEST(61) Multiple of apparent velocity regression r.m.s. residual at which arrival times are weighted to zero during start location determination, used in starting location of **distant** events. Reducing this value will cause arrivals to be rejected when they do not conform to the plane wave set of arrivals which is characteristic of distant events. Unless you are getting a lot of messages ' xxx removed: Apparent velocity deviation =..', in the output, I recommend against changing this default value However, you can disable this feature by setting TEST(61)=0.0.

default: TEST(61)=2.0 (enabled)

TEST(62) = 0.0 limits the IASP91 phase list to "basic", whereas 1.0 specifies all phases to be calculated each time. This can speed up the program on a slower machine.

default: TEST(62)=1.0 (all phases)

TEST(63) Value that specifies the types of phases used by the subroutine TRTIME to be specified when calculating travel times, their derivatives and the resulting residuals and r.m.s. residual.

TEST(63)	Type of Phases Used by TRTIME
0	match 8 character input phase ID (2nd letter is case insensitive)
1	use minimum time for 1st letter of input phase ID
2	use refracted 'n' phases only
3	use surface 'g' phases only
4	use Conrad 'b' phases only
5	use only minimum time P and pP phases (all *s* and *S* first letter phases are weighted out)

default: TEST(63)=0.0 (match phases)

TEST(64) Allows temporary increases in r.m.s. residual, σ. If $\sigma <$ TEST(64)σ_0, where σ_0 is the current minimum value, iterations will proceed as if σ had decreased, i.e., the change in the hypocenter will be allowed. To disable this feature, set TEST(64)=0.0. This was how HYPOCENTER originally worked, but experience with deep events has indicated that secondary minima can exist at shallow depth in some situations. A factor of 2 increase in σ for a single step was found to be sufficient to allow movement to a lower σ at greater depth.

default: TEST(64)=2.0 (σ increase enabled)

TEST(65) Number of iterations for which increases in r.m.s. residual will be allowed.

default: TEST(65)=3.0

TEST(66) Setting TEST(66)=1.0 enables output in PRINT.OUT and on the screen of travel time calculation errors (critical distances exceeded, etc.).

default: TEST(66)=0.0 (no error output)

TEST(67) Setting TEST(67)=1.0 forces blank phases to be recognized as first-arrival P's. Otherwise blank phases are ignored.

default: TEST(67)=0.0 (blank phases ignored)

TEST(68) Apparent $P-$velocity (km/sec) used to calculate starting depth from *PP-P* arrival time differences.

default: TEST(68)=5.0

TEST(69) Distance (deg.) beyond which *PKiKP* or *PKP* are used as the first arrival instead of *Pdiff*.

default: TEST(69)=110.

TEST(70) Maximum depth (km) that the hypocenter is allowed to move to.

default: TEST(70)=700.

TEST(71) determines if phases in the HYP.OUT file and residual summary in PRINT.OUT are sorted by distance. Set TEST(71)=0.0 to disable this sorting.

default: TEST(71)=1.0 (enable distance sorting)

TEST(72) Setting TEST(72)=1 enables the auto-phase pick feature for **distant** events. The travel-time routine then automatically picks the closest IASP91 phase at the current hypocentral position for all non-zero weight input phases (including blanks). If TEST(72)=0 (the default), TRTIME attempts to match the 8-character phase ID's in the input phase file.
default: TEST(72)=0.0 (auto-pick disabled)

TEST(73) Number of iterations for which the program proceeds using only first-P arrivals, before attempting to auto-phase pick when locating **distant** events. This is only used when TEST(73)=1

default: TEST(73)=3.0

TEST(74) Setting TEST(74)=1.0 enables a printout of the input phase data in PRINT.OUT.

default: TEST(74)=0.0 (input phase print disabled)

TEST(75) M_L parameter, a, in the formula

$$M_L = a \log_{10} A + b \log_{10} \Delta + c\Delta + d \tag{4}$$

using the California standards (*Hutton and Boore*, **1987**), where A is output amplitude (nm) and Δ is hypocentral distance (km).

default: TEST(75)=1.0

TEST(76) M_L parameter, b

default: TEST(76)=1.11

TEST(77) M_L parameter, c

default: TEST(77)=0.00189

TEST(78) M_L parameter, d

default: TEST(78)=-2.09

TEST(79) Minimum number of stations required to attempt a solution

default: TEST(79)=1.0

TEST(80) Minimum number of phases required to attempt a solution (azimuth is counted as a separate phase).

default: TEST(80)=3.0

TEST(81) Setting TEST(81)=0.0 disables location of **local** events.

default: TEST(81)=1.0

TEST(82) Setting TEST(81)=0.0 disables location of **regional** events.

default: TEST(82)=1.0

TEST(83) Setting TEST(83)=0.0 disables location of **distant** events.

default: TEST(83)=1.0

TEST(84) Setting TEST(84)=0.0 disables the ellipticity corrections for IASP91 travel times for **distant** events.

default: TEST(84)=1.0 (enabled)

TEST(85) *A priori* error (sec) in arrival times for a weight of 1 for **local** event hypocentral error calculation. This only becomes significant as the number of phases approaches 4. Otherwise, the r.m.s. residual dominates the error estimate. TEST(91) is the corresponding parameter for distant events.

default: TEST(85)=0.1

TEST(86) Number of degrees of freedom used in estimating TEST(85) for **local** events. Making this number large (> 10000), forces TEST(85) to be used as the standard deviation in the calculation of hypocentral errors and the r.m.s. residual to be ignored. TEST(92) is the corresponding parameter for distant events.

default: TEST(86)=8.0

TEST(87) Confidence level that solution will lie outside the confidence ellipsoid defined by the covariance matrix for both **distant and local** events. I have set the default to correspond to standard error (90% confidence)used by the NEIC.

default: TEST(87)=0.1

TEST(88) R.m.s residual (sec) at which residual weighting is applied for **distant** events. Set TEST(88)=0.0 to disable residual weighting.

default: TEST(88)=10000. (enabled at all times)

TEST(89) Set TEST(89)=0.0 to disable use of depth phases in solution.
default: TEST(89)=1.0 (depth phases enabled)

TEST(90) Setting TEST(90)=0 disables the use of core phases (*PKP*, *SKP*, etc.) for **distant** events

default: TEST(90)=1.0 (core phases enabled)

TEST(91) *a priori* estimate of standard deviation in travel time residuals (sec) for **distant** events used to estimate hypocentral errors for distant events. TEST(85) is the corresponding parameter for local events.

default: TEST(91)=1.0

TEST(92) Number of degrees of freedom used in *a priori* estimate of standard deviation in travel time residuals, TEST(91), for **distant** events. Making this number large (> 10000), forces TEST(91) to be used as the standard deviation in the calculation of hypocentral errors and the r.m.s. residual to be ignored. TEST(86) is the corresponding parameter for local events.

default: TEST(92)=8.0

TEST(93) Setting TEST(93)=1 forces the output longitude, θ, to be always positive, $360 > \theta > 0$. TEST(93)=0.0 (the default) $\Rightarrow -180 > \theta > +180$.

default: TEST(93)=0.0 ($-180 < \theta < +180$)

4.1.2 Station Information

Columns	Format	Description
1-2		Free
3-6	A4	Station name
7-8	I2	Station latitude in integer degrees
9-13	F5.2	Station latitude minutes
14	A1	Latitude direction ("N" or "S")
15-17	I3	Station longitude in integer degrees
18-22	F5.2	Station longitude minutes
23	A1	Longitude direction ("E" or "W")
24-27	I4	Station elevation above sea level in meters
28-33	F6.2	Station delay in seconds (subtracted from observed time)
34-38	F5.2	Station magnitude correction, ΔM

The list of stations can be up to 5000 (the default in HYPPARM.INC) and is terminated by a blank line.

4.1.3 Layered Velocity Model Parameters

The input format for these is:

Columns	Format	Description
1-7	F7.3	P velocity in km/s
8-14	F7.3	Depth to bottom of layer in km
15-21	F7.3	S velocity in km/s
22	A1	"N" signifies Moho and "B" Conrad for N and B phases

This layered model can have up to 150 layers (the default in HYPPARM.INC) and is terminated by a blank line. If the S velocities are left blank, they are calculated using the V_p/V_s ratio given below.

4.1.4 V_p/V_s ratio, Starting Depth, Distance Weighting

The input format for this line is:

Columns	Format	Description
1-5	F5.0	Starting depth in km
6-10	F5.0	XNEAR for distance weighting
11-15	F5.0	XFAR for distance weighting
16-20	F5.2	V_p/V_s ratio

The starting depth given here is used in the following cases:

1. An "S" does not appear in column 44 of the phase header line
2. The starting depth on the phase header line is blank
3. TEST(31)=-2

The V_p/V_s ratio is only used if V_s velocities are left blank in the format given in **2.1**.

Distance weighting, which is only used for local events, is specified using the two parameters x_{near} and x_{far}, in km. The distance weight, w_d, is then calculated using

$$w_d = \frac{x_{far} - \Delta}{x_{far} - x_{near}} \tag{5}$$

where Δ is the epicentral distance in km.

4.1.5 Reporting Agency

This can be entered as a 3-letter nmemonic in the left three columns of the line following the V_p/V_s line. It is written on the phase header line in HYP.OUT for each event located using this station file.

4.1.6 Parameter File Example

Below is the example of a station parameter file which is included in the distribution package (STATION0.HYP). The TEST parameter list includes all of those parameters that are likely to be changed in routine location of both local and distant events.

```
RESET TEST(07)=-3.0                  duration magnitude coefficients
RESET TEST(08)=2.6                       "        "
RESET TEST(09)=0.001                     "        "
RESET TEST(13)=5.0                   change in km for minimum r.m.s. test
RESET TEST(31)=3.0                   degrees of freedom (2 to fix depth)
RESET TEST(36)=0.0                   enable local residual weighting (0 disables)
RESET TEST(40)=0.0                   depth origin (0=sea level)
RESET TEST(41)=20000.0               maximum epicentral distance allowed in km
RESET TEST(43)=5.0                   maximum rms for inclusion in residual summary
RESET TEST(51)=3.6                   L phase velocity in km/s
RESET TEST(52)=10.0                  azimuth error used to weight azimuths
RESET TEST(56)=1.0                   enable start location (0 disables)
RESET TEST(60)=1.0                   enable consistency test (0 disables)
RESET TEST(63)=0.0                   type of travel time: 5 forces 1st arrivals
RESET TEST(71)=1.0                   enable output sorting in order of distance
RESET TEST(72)=0.0                   enable autophase pick
RESET TEST(85)=0.1                   a priori standard deviation in local event times
RESET TEST(86)=8.                    degrees of freedom in test(85)
RESET TEST(87)=0.1                   alpha for confidence ellipsoid
RESET TEST(88)=10000.                enables residual weighting for distant events
RESET TEST(89)=1.0                   enable use of depth phases in solution
RESET TEST(90)=1.0                   enable use of core phases
RESET TEST(91)=1.0                   a priori error in distant times

  JMI 7055.70 00843.85W 211
  JNE 7059.39 00817.81W  57
  JNW 7101.72 00825.69W  95

  6.2       0.0
  6.6      12.0
  7.1      23.0
  8.05     31.0       N
  8.25     50.0
  8.5      80.0
```

```
15.0 1000.3300. 1.74
BER
```

4.1.7 Station Elevation Corrections

The default depth origin is set to be sea level, which is always used by the IASPEI software, since the model is a sphere having a radius of 6371 km, i.e., the earth's mean radius. HYPOCENTER maps the station latitudes and longitudes, which are assumed to be geographic, into angles (radians) on the geocentric sphere (radius=6378.2 km) using two routines written by Bruce Julian. The internal coordinates used by HYPOCENTER are then radians, north of the equator and east of the Greenwich meridian positive, and depth positive down (depth is kept in km).

Each station's elevation correction is made by using the horizontal slowness, P_h, (the horizontal partial derivative calculated by the IASPEI software) and a constant velocity layer above sea level having the velocity V, specified for the first layer in the layered model (P and S). Using the horizontal and total slownesses, the vertical slowness, P_z, in the upper layer, can then be calculated using

$$p_z = \left(\frac{1}{V^2} - p_h^2\right)^{\frac{1}{2}} \tag{6}$$

The time to traverse the upper layer, which is added to the IASP91 time is then $P_z h$, where h is the station elevation in km.

For local events, it is possible to set the depth origin to the maximum elevation station, by setting TEST(40)=1.0. As well as allowing hypocentral depths above sea level, this also allows the velocity model to have an arbitrary number of layers down to sea level, as the layered velocity model origin is also at the maximum elevation station. The layered model travel time subroutine (DTDX2) was modified to allow the case where stations are below the hypocenter that they locate, a situation that many previous programs excluded as "air quakes".

4.2 The Phase Data File

4.2.1 General Description

The phase data is stored in a second file which is either part of the SEISAN data base system, or a single file containing a single or multiple events. In the latter case the program prompts the user for the name of the file, which is only constrained by the operating system being used (e.g., in MS-DOS 8 characters, a period and a 3 character extension) HYPOCENTER has been modified to read phase data in the NORDIC format described by *Havskov*, **1990**. This format is now widely used by networks employing the SEISAN software package (*Havskov and Lindholm*, *1994* of which HYPOCENTER is also a part. Since it was more practical to design the program to read a single input format, a number of filtering programs have been written to convert other standard input formats to the NORDIC format. These programs, HYPNOR, HINNOR and ISCNOR, are described in Appendix I.

The phase data format is used for both input and output, enabling a single file to be maintained which contains both the most up to date solutions and their residuals along with the input data. The program does not overwrite the original input data, but generates a separate output file (HYP.OUT) containing both the original input phase data along with the solution and residuals. For plotting purposes, I have included a program COMPRESS which removes all the phase data (type 4) from such a file leaving only the hypocentral solutions (type 1).

4.2.2 Phase Header Line Format

Columns	Format	Description	Comments
1		Free	
2-5	I4	Year	
6		Free	
7-8	I2	Month	
9-10	I2	Day of month	
11	A1	Location indicator	"F"=fix origin time
12-13	I2	Hour	
14-15	I2	Minutes	
16		Free	
17-20	F4.1	Seconds	
21	A1	Velocity model	"X" means read STATIONX.HYP
22	A1	Distance indicator	"L"=local "R"=regional "D"=distant
23	A1	Event ID	"E"=explosion, "V"=volcanic, etc.
24-30	F7.3	Latitude (deg.)	
31-38	F7.3	Longitude (deg.)	
39-43	F5.1	Depth (km)	
44	A1	Depth indicator	"F"=fix depth, "S"=start depth
45	A1	Epicenter indicator	"F"=fix epicenter, "S"=start epicenter "N"=use closest station start location
46-48	A3	Epicenter reporting agency	
49-51	I3	Number of stations used	
52-55	F4.1	r.m.s. residual	
56		Free	
57-59	F3.0	Magnitude #1	
60	A1	Type of magnitude	"L"=ML, "B"=MB, "S"=MS, etc.
61-63	A3	Magnitude reporting agency	
64		Free	
65-67	F3.0	Magnitude #2	
68	A1	Type of magnitude	"L"=ML, "B"=MB, "S"=MS, etc.
69-71	A3	Magnitude reporting agency	
72		Free	
73-75	F3.0	Magnitude #3	
76	A1	Type of magnitude	"L"=ML, "B"=MB, "S"=MS, etc.
77-79	A3	Magnitude reporting agency	
80	A1	Type of this line ("1")	

This type 1 line must be the first line preceding each event.

4.2.3 Phase Data Line Format

Columns	Format	Description	Comments
1		Free	
2-5	A4	Station name	blank⇒end of phase data
6	I1	Weight†	HYPO71 style: "0"=full weight, "4"=zero, "9"=P-S diff.
7	A1	Instrument type	"S"=SP, "I"=IP, "L"=LP
8	A1	Component	"Z", "N", "E"
9		Free	
10	A1	Quality indicator	"I", "E", etc.
11-18	A8	8-character Phase ID†	"PN", "PG", "SKP", etc.
19-20	I2	Hour	can be $>$ 24
21-22	I2	Minute	
23		Free	
24-28	F5.0	Seconds	
29		Free	
30-33	I4	Duration (sec)	
34-40	F7.1	Amplitude (nm)	zero to peak
41		Free	
42-45	F4.0	Period (sec)	
46		Free	
47-51	F5.0	Azimuth (deg.)	relative to station
52		Free	
53-56	F4.0	Phase velocity (km/s)	
57-60	F4.0	Signal to noise ratio	
61-63	I3	Azimuth residual (deg.)	
64-67	F4.1	Travel time residual (sec)	
68		Free	
69-70	I2	10 X Weight	weight used in solution
71-75	I5	Epicentral distance (km)	
76		Free	
77-79	I3	Azimuth at source (deg.)	
80	A1	"4" or blank	

†The program also recognizes the older 4-character format, necessary when polarities are used for fault-plane solutions, etc. This is identical to the above format except in the columns defined below

Columns	Format	Description	Comments
6	A1		MUST be blank for this format
11-14	A4	Phase ID	
15	I1	weight	same as in column 6 above
16		Free	
17	A1	First motion	"U","D":clear, "+","-":unclear

The phase arrival times are all referred to the year, month and day on the type 1 header line preceding them, i.e., for an unlocated event, the year, month and day of all the arrivals must appear on the type 1 record immediately before them. The phase list can be up to 2000 phases (the default in HYPPARM.INC) and is terminated by a blank line (another type 1 record will then define a new location).

4.2.4 Phase File Example

An example of the phase input format for a single event appears below. This event is in the file JMYEX.INP (included in the distribution package) and is the input used to generate all subsequent output examples, along with the parameter file STATION0.HYP, listed in **4.1.5**.

```
1994  117 0335 16.4 L  71.036  -6.524 10.9  BER  3 0.0 2.7CBER                1
STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR AR TRES W  DIS CAZ7
JNE  SZ EP        335 26.87   66                                0.110   65 265
JNE  SZ ES        335 34.31                                     0.010   65 265
JNW  SZ EP        335 27.48   55                                0.010   69 268
JNW  SZ ES        335 35.58                                     0.010   69 268
JMI  SZ ES        335 38.40                                     0.010   81 260
JMI  SZ EP        335 29.06   64                               -0.110   81 260
```

Note that this phase header line already records a previous location, i.e., it was generated as HYP.OUT by a previous location run. This demonstrates the positions of starting epicenter and starting depth if these are used (not in this case). For new phase data, only the date, month and day (columns 2-10) are essential, (see 4.2.2 for details).

4.2.5 Use of S-P and L-P differences

Uncertainty in absolute times often makes it necessary to be able to use the difference in time between two arrivals such as P and S or P and L. In such a case, the corresponding derivative and travel time differences are used in the inversion, and the time residual is left uncentered. If no absolute times at all are available, the calculated origin time will be close to that at the first arrival station and is completely meaningless. However, a perfectly good epicenter and depth can still be obtained from *S-P* or *L-P* differences alone.

To enable a difference phase, set the weight for the P phase input record to 9. This *P* phase is then assigned a weight of 0, effectively disabling its use. However, a time residual and azimuth, etc., will still be calculated for it, enabling an assessment to be made of its absolute time. A search will then be made of the entire input phase set for an *S* or *L* phase at the same station. If such a phase is found, its variables are used to store the observed and calculated difference times and their derivatives, while its weight (0-4) is used for the difference phase (DON'T SET IT TO 9!). If two or more such phases (e.g., S_n, S_g, L_g, etc.) are found, all their differences with the *P* time will be used instead of their absolute times. Blanks will appear beneath 'hrmn' in the residual summary for all such phases, while the observed and calculated difference times with the first *P* will appear beneath 't-obs' and 't-cal' in the residual summary in PRINT.OUT.

Note: Do not attempt to set the weight of more than one phase to 9 at a single station.

4.2.6 Starting Location Options

If an "S" appears in column 44 of the phase header line, the location starts at the epicenter given on the same line (columns 24-38). If an "N" appears in column 44, the starting location epicenter is taken 0.2 km NE of the closest station. If an "S" appears in column 45 on the phase header line, the depth appearing in columns 39-43 will be used as the starting depth (provided TEST(31) is not -2). Using "F" instead of "S" in columns 44 and 45 fixes the epicenter and depth, respectively during all iterations. A "F" in column 11 fixes the origin time. If a "*" appears in column 44, the event is not located, i.e., the location is kept unchanged in HYP.OUT.

Epicenter, depth and origin time locations can also be fixed for a whole input data set by using TEST(31), as described in **4.1.1**. The value of TEST(31) overrides any settings present on the phase header record.

If no starting location is specified and TEST(56)=1.0, HYPOCENTER will use the starting location subroutine STARTLOC (described in more detail in **8.2**). This tests the r.m.s. residuals of all candidate starting locations and selects the minimum r.m.s. residual solution. The starting location depth is determined using depth phases (provided TEST(89)=1.0) if at least two are present, or is fixed at the value either in the STA-

TIONx.HYP file, or the value on the phase set header record (Nordic format) if 'S' or 'F' are specified (described in **4.2.4**).

4.2.7 Travel Time Options

If the distance indicator in column 23 of the phase header record (see **4.2.2**) is "L", a crustal model is used regardless of distance, whereas if it is "D", IASP91 is used. If the distance indicator is "R" or blank, the parameter TEST(57) is used to determine whether a layered model or IASP91 software is used to calculate the travel times and their derivatives. The distances from each station are calculated at each iteration and IASP91 is used if any of them exceed TEST(57).

Warning: Setting the distance indicator to "R" or blank may give distant solutions for local events, particularly if bad readings are present.

4.2.8 Weighting Options

The following weights are used to calculate the solution:

1. **User specified weights,** w_0**:** These are calculated using the HYPO71 style weight number 0 to 4, read with each phase (in either column 6 or 15 of the phase data), where 0 corresponds to $w_0 = 1.0$, 1 to $w_0 = 0.75$, 2 to $w_0 = 0.5$, 3 to $w_0 = 0.25$ and 4 to $w_0 = 0$.

2. **Distance weighting:** This is given by the formula

$$w_d = \frac{x_{far} - \Delta}{x_{far} - x_{near}} \tag{7}$$

   where Δ is the distance of the event from the station and x_{near}, x_{far} are read from the station file, STATION0.HYP, as described in **4.1.4**.

3. **Bisquare weighting:** This scheme, described by *Anderson*, **1982**, calculates residual weights using an $M1$ scheme, described in **4.1.1**. This weighting is not applied until the calculated r.m.s. residual of the solution falls below TEST(36), for local events and below TEST(88), for distant events. The default value of TEST(36) is 0, which disables the weighting for local events. The default value of TEST(88) is 10000, enabling it at all times for distant events.

4. **Azimuth weighting:** Azimuth residuals are divided by TEST(52), which is the azimuthal error that corresponds to a one second error in arrival time. For example,

if TEST(52)=10 (default), a phase residual of 10 degrees will become a residual of 1 second (10/TEST(52)) in the parameter corrections and r.m.s. residual calculation.

Weights (1)-(3) are multiplied together to calculate the travel time weight used in the inversion. If the initial user-specified weight, w_0, is changed by (2) or (3), set to zero by the consistency check, or set to -1 because the phase is not recognized, an asterisk will appear after the final weight in the residual printout.

To calculate the r.m.s. residual, the sum of the squared residuals is divided by the sum of the weights, rather than by the number of non-zero phases. This prevents the absolute magnitude of the weights from lowering the r.m.s. residual.

4.3 Magnitude Calculations

Coda Magnitude

This is calculated using

$$M_C = a + b \log_{10} T + c\Delta + \Delta M \tag{8}$$

T is the coda length in seconds, Δ is the epicentral distance in km and ΔM is the station magnitude correction (see **4.1.2**) and the constants a, b and c are the parameters TEST(7), TEST(8) and TEST(9), respectively.

Amplitude Magnitude

This is defined using

$$M_L = a \log_{10} A + b \log_{10} \Delta + c\Delta + d \tag{9}$$

using the California standards (*Hutton and Boore*, **1987**), where A is output amplitude (nm) and Δ is hypocentral distance (km) and a, b, c and d are the parameters TEST(75), TEST(76), TEST(77) and TEST(78), respectively.

Body Wave Magnitude

This is defined using

$$M_B = \log_{10} \frac{A}{T} + Q(\Delta, d) \tag{10}$$

where A is output amplitude (nm) and T is period (sec). $Q(\Delta, d)$ is a distance-depth correction factor, which is interpolated from the table given by *Veith and Clawson*, **1972**.

4.3.1 Surface Wave Magnitude

This is defined using

$$M_S = \log_{10} \frac{A}{1000T} + 1.66 \log_{10} \Delta + 3.3 \qquad (11)$$

where A is amplitude (nm), T is period (seconds) and Δ is the epicentral distance (degrees). The constants in this equation can only be changed by editing HYPOSUB3.FOR and recompiling the program.

5 USING THE PROGRAM

The program is started by the command "hyp". The program then reads the global travel time direct-access file IASP91.HED and the station parameter file STATION0.HYP. The user is then prompted for an input file and whether or not to run in interactive mode. If the program is running in the SEISAN environment, additional questions will be asked (date, time), see the SEISAN manual for details.

For non-interactive input, the program will locate all events in the input file without any more questions and one line with the location will appear on the screen for each event (see example below). In the case of interactive input, the program will print a header line for the first event and wait for a response. The response at this point can be an event number, in which case the program will read down (or back) through the input file to that event number and print its header on the screen. The command "l" (or "L") will locate the current event and print the solution, then the residuals on the screen. After printing all the residuals, entering a comma (",") will print several lines of the PRINT.OUT file for that event, including the difference from the solution on the event's header record. Repeating the comma command will print more lines of PRINT.OUT. Entering a period (".") will print previous lines of PRINT.OUT. The comma and period can therefore be used to interactively examine all the details for the current solution. Entering "q" (or "Q") will generate a summary for all the events located and quit the program.

For each event, a new parameter file can be read in if needed by specifying a letter other than blank in column 21 of the phase header line, as described in **4.2**. HYPOCENTER then determines the start location using STARTLOC, and providing there are sufficient phases, iterates to find a solution. Either the layered velocity model or the IASPEI global model is used to determine travel times and their derivatives, the remaining calculations being identical. The operator can specify whether the event is local, regional or distant in column 22 of the event's header record (see 4.2.6). Once a solution is obtained, magnitudes are calculated and the results are printed out in the output files, PRINT.OUT and HYP.OUT, as well as on the screen when in interactive mode. At the end of a sequence of event locations, the mean residuals and r.m.s. deviations for all the stations used are printed out.

An example of the screen dialog for interactively locating the event in the file JMYEX.INP is shown below:

```
Filename for file in NORDIC format       :

jmyex.inp

Interactive operation (N/Y=default)Y

#    1  1994  117 0335 16.4 L NPHS=   6   T Q L #XXX L

  date hrmn   sec      lat      long depth   no m    rms  damp erln erlt erdp i
94 117  335 16.60 7059.49N   6 36.5W  23.6    6 3    .04  .000 48.9 58.3352.0
stn   dist   azm  ain w phas    calcphs hrmn tsec  t-obs  t-cal    res   wt di
JNE     61 269.0 92.6 0 P       PG       335 26.9   10.3   10.2    .06 1.00  8
JNE     61 269.0 92.6 0 S       SG       335 34.3   17.7   17.8   -.06 1.00  9
JNW     66 272.7 91.9 0 P       PG       335 27.5   10.9   10.9   -.01 1.00 14
JNW     66 272.7 91.9 0 S       SG       335 35.6   19.0   19.0    .02 1.00 25
JMI     78 263.8 91.1 0 S       SG       335 38.4   21.8   21.8    .04 1.00 29
JMI     78 263.8 91.1 0 P       PG       335 29.1   12.5   12.5   -.05 1.00 15

1994  117 0335 16.6 L  70.991  -6.608 23.6  BER  3  .0 1.7CBER

#    1  1994  117 0335 16.6 L NPHS=   6   T Q L #XXX Q
  print output in file print.out
 CAT-file in file hyp.out
 Summary file in hypsum.out
Stop - Program terminated.
```

The first prompt is for a Nordic phase format filename containing a set of events to be located. The next prompt, "Interactive Operation" determines whether all the events in the file are located ("N") or interactively selected ("Y" or return, as in this example). The next line of output lists the first event number, origin time, local/distant flag and number of phases (NPHS). The following letters then determine the program's action

1. **A number** moves to that event number in the file (can be back or forward).

2. **T** prints out the input phase data.

3. **L** locates the event and prints the residual summary in the PRINT.OUT file (as in the example).

4. **, or .** moves forward and backward, respectively, in the PRINT.OUT file, printing the lines on the screen.

5. **Q** quits the program and generates a summary in PRINT.OUT.

6 PROGRAM OUTPUT

6.1 Description of Output Files

HYPOCENTER generates the following output files:

PRINT.OUT	detailed output with all iterations etc.
HYP.OUT	output in Nordic format
HYPSUM.OUT	summary output in HYPO71 format

If TEST(71)=1 (the default), both the station residuals in PRINT.OUT and the output phase lines in HYP.OUT are sorted with respect to epicentral distance. The order of multiple phases at each station is left unchanged, as is the order of the input header lines.

These files are overwritten during each run, so care must be taken to rename old outputs if they are needed. APPENDIX III contains an example of the PRINT.OUT file obtained by running HYPOCENTER on the input files EXAMPLE.DAT and STATION0.HYP.

6.2 Criteria for a Solution:

The cases where a solution will either not be attempted, or depth will be fixed, are as follows:

1. Multiple phases at two stations, but no azimuths

   This is a non-unique case, even though four different arrivals are present.

2. Three or less phases at three different stations and no azimuths

   HYPOCENTER will calculate a fixed-depth solution (at the starting depth in STATIONX.HYP) for three phases at different stations (this case was previously excluded).

3. A single phase at one station with an azimuth

4. The user has specified more stations (using TEST(79)) and/or phases (using TEST(80)) than are available in the phase set.

6.3 The PRINT.OUT File

6.3.1 Sample PRINT.OUT File

```
Input File: jmyex.inp

Reset test( 7)=   -3.0000
Reset test( 8)=    2.6000
Reset test( 9)=     .0010
Reset test(13)=    5.0000
Reset test(31)=    3.0000
Reset test(36)=     .0000
Reset test(40)=     .0000
Reset test(41)=20000.0000
Reset test(43)=    5.0000
Reset test(51)=    3.6000
Reset test(52)=   10.0000
Reset test(56)=    1.0000
Reset test(60)=    1.0000
Reset test(63)=     .0000
Reset test(71)=    1.0000
Reset test(72)=     .0000
Reset test(85)=     .1000
Reset test(86)=    8.0000
Reset test(87)=     .1000
Reset test(88)=10000.0000
Reset test(89)=    1.0000
Reset test(90)=    1.0000
Reset test(91)=    1.0000

    --------- HYPOCENTER Version 3.2  1994 -----------

Maximum elev. station: JMI     7055.70    843.85W   elevation =  211 m

Trial depth = 15.00  Vp/Vs =  1.74

Velocity Model

Depth, km   Vp, km/s   Vs, km/s
      .00       6.20       3.56
    12.00       6.60       3.79
    23.00       7.10       4.08
    31.00       8.05       4.63   N
    50.00       8.25       4.74
    80.00       8.50       4.89

EVENT #      1

1994  117 0335 16.4 L  71.036  -6.524 10.9  BER  3 0.0 2.7CBER                 1
```

```
Consistency check performed

  First arrival: JNE  94 117    3 35 26.87

Starting location depth =    15.0 km
Starting location .15 km from closest station:        70.95   -8.22
rms =     2.93

maximum multi-station phase: P          3

Regression azimuth=  86.5 Apparent velocity=   7.40 km/s

delta =    1.0 R2 =   1.00
Starting location from regression azimuth and slowness:        70.90   -5.43
rms =     2.33

  3 multiple-phase stations

       JNE   P -S       80.9 km
       JNW   P -S       88.1 km
       JMI   P -S      101.6 km

Starting location from 2 distances:   71.68   -7.93
rms =     1.53
Starting location from 2 distances:   71.06   -6.04
rms =     1.25

iter   origin      lat      long depth  no  m    rms    damp.  erlg  erlt  erdp
        (sec)  (dg mn)   (dg mn)  (km)          (sec)           (km)  (km)  (km)
   1    12.08 7110.63N   6 2.38W  15.0   6  2   1.25    .005    .0    .0    .0
   2    15.31 7052.28N   626.77W  15.0   6  2    .37    .005  83.1 102.9    .0
   3    16.52 7056.42N   634.69W  15.0   6  2    .07    .005  39.1  40.3    .0
   4    16.60 7056.52N   635.28W  15.0   6  2    .07    .005  32.6  39.8    .0
depth freed: icd=   3
   5    16.48 7057.50N   636.73W  28.2   6  3    .09    .160  40.7  42.4 152.6
   6    16.62 7058.30N   637.57W  27.6   6  3    .07    .107  34.4  55.0  10.3
   7    16.54 7059.20N   636.00W  23.6   6  3    .05    .027  57.4  57.2  86.0
   8    16.71 7059.18N   636.17W  17.2   6  3    .09    .013  47.8  57.4 357.0
   9    16.62 7058.20N   637.81W  28.2   6  3    .07    .036  56.9  50.4 122.9
  10    16.53 7059.19N   635.97W  23.8   6  3    .05    .024  60.3  58.0  84.2
  11    16.64 7059.18N   636.17W  19.3   6  3    .08    .032  47.1  56.6 241.0
  12    16.71 7058.34N   638.40W  27.5   6  3    .07    .021  68.4  53.7 112.8
  13    16.53 7059.15N   635.93W  23.4   6  3    .05    .014  62.3  60.1  92.7
  14    16.53 7059.15N   635.93W  23.3   6  3    .05 153.427    .2    .3   2.2
  15    16.53 7059.16N   635.94W  23.2   6  3    .05 102.285    .3    .5   4.0
  16    16.54 7059.20N   636.00W  23.6   6  2    .05    .005    .5    .7  11.5
  17    16.60 7059.49N   636.49W  23.6   6  2    .04    .005  33.2  45.6  11.5

  date hrmn   sec      lat      long depth   no m    rms  damp erln erlt erdp ic
```

```
94 117  335 16.60 7059.49N   6 36.5W  23.6    6 3    .04  .000 48.9 58.3352.0  3

Origin time error:       1.44

DRMS Values: d=    5.00 km

DRMS:       lon+d     lon-d     lat+d     lat-d   depth+d   depth-d
 DRMS pos     .22       .22       .03       .03       .04       .05

Resolution matrix: k =       .005

                            Long      Lat   Depth
                    Long    .990     .004    .007
                     Lat    .004     .992   -.005
                   Depth    .007    -.005    .987

Azimuthal Gap in Station Coverage 351 degrees

stn   dist   azm  ain w phas    calcphs hrmn tsec  t-obs  t-cal    res   wt di
JNE     61 269.0 92.6 0 P       PG       335 26.9   10.3   10.2    .06 1.00  8
JNE     61 269.0 92.6 0 S       SG       335 34.3   17.7   17.8   -.06 1.00  9
JNW     66 272.7 91.9 0 P       PG       335 27.5   10.9   10.9   -.01 1.00 14
JNW     66 272.7 91.9 0 S       SG       335 35.6   19.0   19.0    .02 1.00 25
JMI     78 263.8 91.1 0 S       SG       335 38.4   21.8   21.8    .04 1.00 29
JMI     78 263.8 91.1 0 P       PG       335 29.1   12.5   12.5   -.05 1.00 15


Difference from previous solution:

dorigin=      .2 sec dx=    -3.1 km dy=    -5.0 km dz=    12.7 km drms=   .04

unweighted rms =      .04

-----------------------------------------------------------------------------

average differences from previous solutions:        1 events

                           mean         rms

origin time:                 .2 sec        .0
longitude:                 -3.1 km         .0
latitude:                  -5.0 km         .0
depth:                     12.7 km         .0

average station residuals (weighted):

station     p residual rms dev  no p     s residual rms dev  no s
JMI              -.05     .00     1            .04     .00     1
JNE               .06     .00     1           -.06     .00     1
```

```
 JNW                -.01     .00     1             .02     .00     1

 Mean rms value (   1 events) =         .043
```

6.3.2 Input Phase and Station Description

The first line of PRINT.OUT indicates the name of the input phase file that is being processed. This is followed by the RESET TEST parameter list, then by the maximum elevation station and whether it is used as the depth origin.. This is followed by the trial depth, V_p/V_s ratio and the velocity model in STATIONX.HYP. After this, the output described below is repeated for each event. However, the above output is regenerated if a new STATIONX.HYP file is read in.

6.3.3 Event Description

The event number corresponding to the number on the interactive screen dialog is printed (events are numbered sequentially from the start of the file regardless of whether or not they can be located). The phase data header record is printed and if TEST(74)=1.0 (not the default), the input phase arrival times. A message then indicates if the consistency check was performed on these arrival times. Information concerning the starting location follows. The location 0.2 km NE of the first-arrival station is always listed first, along with its r.m.s. residual. If the starting location feature is enabled, depth phase information, regression azimuth and apparent velocity, multiple phase distances are listed, along with all candidate starting locations and their r.m.s. residuals.

6.3.4 Iteration Steps

Results for individual iterations are now printed. Iterations commence at either the closest station or, if starting location is enabled, at the start location having the minimum r.m.s. residual. Each line of output then displays the iteration number, the new origin time (relative to the closest minute), the epicenter, depth, number of non-zero weight phases (including azimuths) used ("n"), the degrees of freedom ("m"), the r.m.s. residual, damping and errors in horizontal position ("erlg") and the depth ("erdp"). Additional messages include when depth was freed, etc. If any non-zero weight "n" phase is present in the input data, the hypocentral depth will not be allowed below the base of the crust as defined in the layered velocity model (see **4.1.3**). When iterations cease, a final line containing the complete origin time and the values above for the minimum r.m.s. residual solution are printed, along with the convergence criteria, IC, described below.

6.3.5 Criteria for Convergence, IC

The value "IC", listed with last line of the iterative output, is the convergence criterion used to stop the iterations. Its value is defined as follows:

IC=1	The r.m.s. residual is less than 0.01 seconds
IC=2	The damping was increased more than TEST(37) times to prevent an increase in the r.m.s. residual
IC=3	The magnitude of the hypocentral change is less than TEST(32) km.
IC=6	Less than three non-zero weight phases as a result of weighting (no solution)
IC=7	More than 10 attempts were made to make the depth negative, or to exceed the maximum parameter change given by TEST(2)

IC will normally be 3, as in virtually all problem situations, the damping is increased to the point where the hypocentral changes become very small.

6.3.6 Minimum R.M.S. Residual Test

If TEST(13)> 0 (the default is 20 km), the changes in the r.m.s. residual which result from moving the hypocenter this distance either side of the solution are printed out ("DRMS Values"). These are very useful for determining if the final solution reached a true minimum in the r.m.s. residual. The line after "DRMS Values" contains either the notation "DRMS pos", indicating that a minimum was attained, or "DRMS neg" indicating that it was not. A global search for "DRMS neg" is often a good way of finding questionable solutions. Experience with the program has indicated that residual weighting can cause the DRMS values to become negative. At this point the least-squares rationale has been thoroughly violated, particularly when extremely large outliers exist in the solution, so this result is not surprising.

6.3.7 Hypocentral Errors

Least-squares error analysis (e.g., *Flinn*, **1965**) gives the equation of the error ellipsoid as the vectors $\mathbf{\Delta x'}$ satisfying

$$(\mathbf{\Delta x'} - \mathbf{\Delta x^0})^T (\hat{\mathbf{G}}^T \hat{\mathbf{G}})(\mathbf{\Delta x'} - \mathbf{\Delta x^0}) \leq \kappa_\alpha^2 \tag{12}$$

where $\mathbf{\Delta x}^0$ is the least-squares solution, $\hat{\mathbf{G}}$ is the partial derivative matrix $\mathbf{G}$, with each of its rows normalized by dividing by the error in the corresponding row of $\mathbf{\Delta t}$. κ_α^2 is given by

$$\kappa_\alpha^2 = m\hat{s}^2 F_\alpha(m, n-m) \tag{13}$$

with $F_\alpha(m, n-m)$ being the F-distribution having m and $n-m$ degrees of freedom at a confidence level α ($m = 4$ for a full solution, 3 for a fixed depth solution, etc.) and $\hat{s}^2$ is the normalized variance of $\mathbf{\Delta t}$. $\hat{s}^2$ is estimated *a posteriori* from the normalized travel time residuals, $\hat{\mathbf{e}}^\mathbf{0}$ using

$$\hat{s}^2 = |\hat{\mathbf{e}}^\mathbf{0}|^2/(n-m) \tag{14}$$

We have taken the error in the origin time, Δt_0, to be the confidence interval in the arrival times, i.e.,

$$\Delta t_0 = \hat{s}^2 F_\alpha(m, n-m) \tag{15}$$

It could be argued that in taking the mean of a set of normally distributed residuals in equation (14), we have reduced the error in this mean by a factor of $\sqrt{(n-m)}$. However, the travel time residuals are in many cases due to heterogeneity in the seismic velocity structure. We therefore cannot automatically assume that by taking more data we reduce the variance in their mean, especially since we are using them to estimate their own variance!

Evernden, **1969** has pointed out that equation (13) gives unrealistically large errors when n is close or equal to m (infinite when $n = m$). He proposed using

$$\kappa_\alpha^2 = \chi_\alpha^2(m) \tag{16}$$

effectively assuming that the errors of the $\mathbf{\Delta t}$ are perfectly known. However, the largest errors in earthquake location are often the errors in the assumed velocity model, which are difficult to determine. We have therefore used the method of *Jordon and Sverdrup*, **1981**, who effectively combined both equations (2) and (4) using the equation

$$\kappa_\alpha^2 = m\hat{s}^2 F_\alpha(m, k+n-m) \tag{17}$$

where

$$\hat{s}^2 = \frac{k + |\hat{e}^0|^2}{k + n - m}. \tag{18}$$

k is then the number of degrees of freedom used to determine the *a priori* errors in the travel times, which are also used to normalize the partial derivative matrix, $\mathbf{G}$. Equations (2) and (4) then appear as limiting cases ($k = 0$ and $k = \infty$, respectively) of equation (17)
.

Jordon and Sverdrup suggested a value of $k = 8$ for teleseismic location, which they showed was equivalent to assuming an *a priori* error of 25% in the normalized travel time residuals. We have added the variable k as an input parameter, allowing it to be easily changed from a default value of 8 to give either Flinn's, **1965** method ($k = 0$) or *Evernden's*, **1969** method ($k \approx 10^{33}$). We also specify the confidence level of the F-distribution as an input parameter (default 0.1, or 90% confidence, the value used by the NEIC),which allows users to set α to other values (e.g., 0.316 or 68.4%) if so desired. We specify the assumed standard deviation in the travel times corresponding to a weight of 1 as input parameters, which are defined separately for local and distant events with defaults of 0.1 and 1 seconds, respectively. This allows us to specify our *a priori* knowledge of errors in the arrival times. It also implicitly defines the errors which correspond to our HYPO71-style phase weights of 0-3 (for our distant default value of 1 seconds, $0 \Rightarrow \pm 1$, $1 \Rightarrow \pm 1.33$, $2 \Rightarrow \pm 2$, $3 \Rightarrow \pm 4$ seconds). It is necessary to specify this absolute error in order to calculate the normalized partial derivative matrix, $\hat{\mathbf{G}}$, and $\hat{\mathbf{e}}^0$ in equation (17).

We can now calculate the elements of the transformed covariance matrix

$$\mathbf{C} = (\hat{\mathbf{G}}^T \hat{\mathbf{G}})^{-1} / \kappa_\alpha^2 \tag{19}$$

The horizontal (e_h) and vertical (e_z) errors are then $e_h = \sqrt{C_{xx}^2 + C_{yy}^2}$ and $e_z = \sqrt{C_{zz}}$ In order to maintain compatibility with the existing HYPO71 summary file format, we output the above two errors in the old format and append the additional four parameters, $\sqrt{C_{xx}}$, C_{xy}, C_{xz} and C_{yz} to the end of each line. This allows reconstruction of the complete covariance matrix in plotting programs.

As *Lahr*, **1978**, has observed the diagonal elements of the covariance matrix can underestimate the total error if two or more parameters are highly correlated,as the diagonal elements define the intersection points of the error ellipse with the x, y and z axes and not the resolved magnitudes of the ellipsoid axes. It is therefore necessary to obtain the eigenvalues and eigenvectors of the covariance matrix to reliably estimate the size of the error ellipsoid in different directions. This is done in a plotting program supplied with HYPOCENTER.

6.3.8 The Resolution Matrix

The resolution matrix indicates the degree of correlation between the three hypocentral coordinates. The parameter corrections, $\mathbf{\Delta x} = (\Delta x, \Delta y, \Delta z)$ are obtained from the partial derivative matrix, $\hat{\mathbf{G}}$, using

$$\mathbf{\Delta x} = \mathbf{A^{-1}} \mathbf{\Delta t} \tag{20}$$

where

$$\mathbf{A} = \hat{\mathbf{G}}^T \hat{\mathbf{G}} + k\mathbf{I} \tag{21}$$

and k is the damping. The resolution matrix is then the (3 x 3) matrix, $\mathbf{R}$, given by

$$\mathbf{R} = \mathbf{A}\mathbf{A^T} \tag{22}$$

$$= \mathbf{V^T}\mathbf{V} \tag{23}$$

where $\mathbf{V}$ is the (3 x 3) eigenvector matrix of $\hat{\mathbf{G}}^T \hat{\mathbf{G}} + k\mathbf{I}$ (*Jackson*, **1972**). Without any damping, k, in the solution, $\mathbf{R}$ will be a diagonal identity matrix. As damping is increased, the off-diagonal terms indicate the degree of correlation. For example, if the longitude was perfectly correlated with depth, the resolution matrix could be indicating that the longitude is negatively correlated with depth, i.e., decreasing the longitude can be offset by increasing the depth, and vice-versa. Instead of a unique solution, the solutions lie along a line, whose direction is defined by the magnitudes and signs of the off-diagonal elements of the resolution matrix. The degree of resolution is the result of the 'trade off' between the degree of resolution of parameters and their variances, which is a part of any inversion. In HYPOCENTER, the parameter controlling this trade off is the damping factor, k. When k is close to its initial value (normally 0.005), the resolution matrix is close to the identity matrix, indicating independent (i.e., unique) resolution of the three hypocentral parameters. As k is increased, the diagonal elements of the resolution become smaller, with a corresponding increase in size of the off-diagonal elements.

The resolution matrix that is printed out by HYPOCENTER has the damping, k, that was used for the minimum r.m.s. residual step, not the damping used to calculate the hypocentral errors.

6.3.9 The Data Information matrix

The $(n \times n)$ data information matrix, $\mathbf{D}$, is defined by

$$\mathbf{D} = \mathbf{A^T A} \tag{24}$$

$$= \mathbf{U^T U} \tag{25}$$

where $\mathbf{U}$ is the $(n \times 3)$ eigenvector matrix of $\hat{\mathbf{G}}\hat{\mathbf{G}}^T + k\mathbf{I}$

Minster et al., **1974** refer to the diagonal elements of $\mathbf{D}$ as *data importances*, as they describe the relative weight given to each of the n residuals, $\mathbf{\Delta t}$, in computing the corrections, $\mathbf{\Delta x}$. The data importances are affected both by the size of the partial derivatives in $\hat{\mathbf{G}}$, as well as by the relative sizes of the residuals, $\mathbf{\Delta t}$. In a well constrained solution, they should all be of roughly equal size ($\approx 100/n\%$). In HYPOCENTER, I have changed the definition of data importance slightly by dividing each of the diagonal elements of $\mathbf{D}$ by the number of degrees of freedom in the solution, m (normally $m = 3$, $m = 2$ for fixed-depth solutions) and converting them to percentages. Since the sum of the diagonal elements of $\mathbf{D}$ is always m, these "normalized data importances directly give the percentage contribution of each data point to the solution.

Like the resolution matrix, the data importances are those for the damping at the minimum r.m.s. residual step, not the damping used to calculate the errors.

6.3.10 Residuals for each Station

The "azimuthal gap in station coverage" indicates the largest sector of azimuth, measured relative to the event, that exists in the station coverage. A large value (e.g., 300 degrees) indicates that the event is far from the center of a small network and the depth values, in particular, are unreliable unless depth phases have been identified.

The residual output headings are defined below:

stn	Station ID
dist	Distance in km
azm	Azimuth at the source
ain	Angle of incidence at the source
w	Input weight (HYPO71 format)
phas	Phase ID
calcphas	phase used to calculate the theoretical travel time
hrmn	Hour minute
t-sec	Arrival time (seconds)
t-obs	Observed travel time (seconds)
t-cal	Calculated travel time for "calcphas"
stdly	Delay (seconds) given in station file
res	Residual (seconds)
wt	Weight used in solution
fmag	Coda magnitude
di	normalized data information

Note: Blanks under "calcphas" and "wt" mean that the phase travel time could not be calculated.

If amplitudes are available, M_B, M_C or M_S will be calculated, and all stations used in the calculation will be displayed at the end of the interactive printout.

An eight character phase ID is used for "calcphas". When the layered model derivatives are used, "calcphas" contains the first phase letter ("P" or "S"). The second letter is "G" if a direct path time was used, "N" if a refraction in the Moho layer was used and "B" if a Conrad layer was used. The "N" and "B" layers must be specified in the layered model input, otherwise the second letter of "calcphas" will be "G" or blank. When the IASP91 software is used, the 8 character IASP91 phase ID is printed under "calcphas".

6.3.11 Residual Summary

In order to assess arrival time delays for P and S at individual stations, it is useful to calculate the mean station residuals for a set of events. This "residual summary" is included at the end of every PRINT.OUT file. Also summarized are the number of residuals at each station and their r.m.s. deviation. It is also possible to exclude poorly fitted events by the use of TEST(43), as only residuals of events having an r.m.s. residual<TEST(43) are used in the summary. The default value of TEST(43) is 15 seconds, allowing virtually all located events to be included.

6.3.12 Difference from Previous Solution

The difference in the solution obtained and the solution on the phase header (if this is not blank) is printed in PRINT.OUT. Averages of the differences in the four hypocentral parameters are also accumulated and their mean values and standard deviations are printed at the end of a run. By using the HYP.OUT file for input, it is then possible to examine statistical differences, due to fixing depth, etc., for a large set of events. I have found it convenient to use a simple run-time compiler, such as QuickBASIC, to extract the difference information from the PRINT.OUT file and plot histograms, etc. I have also set up the ISC conversion program, ISCNOR, to put the ISC locations in the phase header record, allowing easy comparison with the ISC locations.

6.3.13 Other Parameters

Unweighted RMS This is the r.m.s. residual value obtained using all the positive weight (valid) residuals (excluding azimuths) with weights of 1.
Mean RMS This is the r.m.s., $\overline{\sigma}$ of all the individual r.m.s. residual values, σ_i, used in the residual summary (i.e., those with $\sigma_i \leq$ TEST(43) and $>$ 3 valid phases), along with the number of events used to obtain it, i.e.

$$\overline{\sigma} = \left(\sum_{i=1}^{n} \frac{\sigma_i^2}{n} \right)^{\frac{1}{2}}. \tag{26}$$

This is useful for comparing the overall fit of the data for different velocity models, etc.

6.4 The HYP.OUT File

This file has an similar format to the input phase file shown in **4.2.8** and can be used as input for another location. However, the stations are sorted in ascending order of distance from the event (if TEST(71)=1), the type 1 (header) record contains the hypocenter and magnitudes determined by HYPOCENTER and the travel time and azimuth residuals, magnitudes, etc., have been added to each phase record. The hypocentral errors are also placed in the record following the phase header (added if it does not already exist). This error record has the following format:

Columns	Format	Description
15-20	F6.2	Origin time error
25-30	F6.1	Latitude (y) error (km)
33-38	F6.1	Longitude (x) error (km)
39:43	F5.1	Depth (z) error (km)
44-55	E12.4	covariance(x, y) (km^2
56-67	E12.4	covariance(x, z) (km^2
44-55	E12.4	covariance(x, y) (km^2

The HYP.OUT for the example input file, JMYEX.INP, is shown below

```
1994  117 0335 16.6 L  70.991  -6.608 23.6   BER  3  .0 1.7CBER              1
               1.44      58.3    48.9352.0  -.1116E+04  -.1216E+05   .1172E+05E
ACTION:UPD 94-09-19 13:00 OP:bms  STATUS:              ID:19940117033505     I
9401-17-0335-05S.JMI_08                                                      6
STAT SP IPHASW D HRMM SECON CODA AMPLIT PERI AZIMU VELO SNR AR TRES W  DIS CAZ7
JNE  SZ EP        335 26.87   66                                .110   61 269
JNE  SZ ES        335 34.31                                    -.110   61 269
JNW  SZ EP        335 27.48   55                                .010   66 273
JNW  SZ ES        335 35.58                                     .010   66 273
JMI  SZ ES        335 38.40                                     .010   78 264
JMI  SZ EP        335 29.06   64                                .010   78 264
```

For large files, it is useful at this point to run the program COMPRESS to extract the hypocenter records into a smaller file for plotting.

6.5 The HYPSUM.OUT File

This file contains a line for each located event in HYPO71 (summary file) format. This file is useful for analyzing and plotting hypocentral data using existing software packages that read the HYPO71 output format. I now add the additional errors in longitude (format F5.1), the three off-diagonal elements of the covariance matrix and the origin time error (format 4E10.3) to the end of each line in addition to the horizontal (ERH) and vertical (ERZ) errors. As well as maintaining backward compatibility with the old format, this allows reconstruction of the complete covariance matrix and calculation of the hypocentral error ellipsoid. The HYPSUM.OUT file for JMYEX.INP is shown below

Columns 1-78 (old HYPO71 format):

```
Date    Origin    Lat      Long       Depth    Mag No Gap Dmin  Rms  Erh  Erz
```

```
940117 0335 16.60 70 59.49  -6 59.49   23.6    1.7  6 351 61.4  .04 76.1352.0
```

Columns 79-124 (additional errors added):

```
  Erx      Cvxy       Cvxz       Cvyz      Oterr
 48.9 -.112E+04 -.122E+05  .117E+05  .144E+01
```

7 PROGRAM ERROR MESSAGES

The following error messages can appear in the PRINT.OUT file as well as on the screen, depending on the values of the TEST parameters. In all cases, they will not cause the program to stop, it should continue processing subsequent events. In most cases, the phases that the messages refer to will be weighted to zero, so they should not effect the solutions. Some of the messages indicate that the input files have the wrong format, missing stations, etc., and should of course be remedied to get any reliable locations.

All valid P times are inconsistent

The consistency test was unable to find a single pair of non-zero weight arrival times at different stations whose difference was less than the travel time between the same two stations. A similar message can appear for S. Check the input phase data for errors in the arrival times. If the error persists, disable the consistency test by setting TEST(60)=0.

Blank line in file: check !!

A blank line was read in the input PHASE file where it wasn't expected, i.e., not separating events.

character in starting depth field on input phase header: XXXXX

An illegal character was found in columns 39-43 on the input phase header record. The offending field is listed. Edit the input phase file and correct the record: it must be a valid number.

Depth limited to moho

When "n" (refracted) phases are specified as input, HYPOCENTER will not allow the hypocentral depth to go below the "N" layer in the velocity model. Remove the "n" in the input phase ID if you want the solution to proceed below the Moho.

Depth limited to Conrad

A "b" phase was specified and HYPOCENTER limited the depth to above the "B" layer in the velocity model

Depth fixed at XXX.X

HYPOCENTER fixes the hypocentral depth when it reaches TEST(70) km during iterations (the default is 700 km). Increase TEST(70) if you want the solution to go deeper.

DTDX2 error: event below Moho - N phase not possible

The layered model travel time subroutine, DTDX2, was unable to calculate a time for a (requested) "N" phase due to the event being below, or in the refracting layer.

DTDX2 error: N phase not possible in a single layer

Only a single layer was given in the input velocity model, while an "N" phase was requested

DTDX2 error: event coincident with station - no refracted phase

A location caused the event to be at the same location as the station. Could occur for a single station location where P and S phases had identical arrival times.

DTDX2 error: refracted phase not possible in bottom layer

A refracted ("N") phase could not be calculated due to the event being in the bottom layer

DTDX2 error: refracted phase not possible, delta $<$ critical distance

An "N" or "B" phase was requested for an event at less than the critical distance from a station, where no refracted phase was possible.

DTDX2 error: p $>$ 1/velocity in layer x, p = xx.xxx

This error occurred in the direct path travel time calculation. It could be caused by a low-velocity layer being specified.

DTDX2 error: Illegal velocity inversion

A velocity has been specified in a layer that is less than that in the overlying layer.

Event is XXX.X km from closest station: no output

The epicentral distance of TEST(41) km has been reached. Change the value of TEST(41) if you want the iterations to proceed (the default value of 20000 should always prevent this message from occurring)

Event not located, see files PRINT.OUT and HYPERR.OUT

Column 1 of the phase header record contained an "N", indicating that HYPOCENTER was unable to locate the event, probably due to an insufficient number of phases. The PRINT.OUT file should contain details. It may be useful to set TEST(66)=1 to get a printout of travel time calculation errors to see if phases are invalid and are being weighted to zero (see below).

GETPHASN: error in header record

An error was made in interpreting a phase header record in the input phase file. Check that a letter has not been placed where a number should be

GETPHASN: Error in phase record

An error was made in interpreting a phase record in the input phase file. Check that a letter has not been placed where a number should be

Maximum arrivals exceeded: edit HYPPARM.INC and recompile sources

The parameter NARRIV in the include file, HYPPARM.INC has been exceeded by the number of phases for an event. This could be due to missing header records. If the default value has in fact been exceeded (check HYPPARM.INC), it will require changing NARRIV and recompiling and linking the executable (which could also require more RAM on the computer)

Moho phase specified, moho not defined

An "n" (refracted) phase was specified without including an "N" next to the layer in the velocity model defining the Moho. See the PROGRAM INPUT section for details

No absolute times: origin time not calculated

All valid arrival times have either been weighted to 0 or 9. A valid hypocenter may be calculated, but the origin time is invalid.

No DTDX2 travel time for XXXX XXXX

There was no valid phase of the type specified at the current hypocenter position (e.g., an "n" phase at less that the critical distance). The phase is temporarily weighted out and iterations proceed normally.

No IASP91 travel time for XXXX XXXX

There was no IASP91 phase of the type specified at the current hypocenter position (e.g., a "PKP" phase at 60 degrees). The phase is temporarily weighted out and iterations proceed normally.

Station XXXX is not on station list: ignored

The identified station could not be found in the list of stations in the STATIONX.HYP file. None of the phases for this station can be used in the location. Check the STATIONX.HYP file to ensure you have included this station, or that you have the right parameter file.

STOP WITH STATUS XXX

This is a SEISAN message relating to the SEISAN database access. See the SEISAN manual for details.

XXXX phase XXXX weighted out: inconsistent time

A phase at the specified station failed the consistency test and was weighted to zero. Check its residual in the output to look for a misread minute, hour or day. Otherwise, disable the consistency check.

XXXX XXXX phase removed: Apparent velocity deviation = XXXX.XX

During the start location determination, a phase was found whose deviation from the plane wave approximation was more than TEST(61) times the overall r.m.s. residual from the plane wave solution. Serious outliers such as this can dramatically change the azimuth and apparent velocity of distant events and are therefore weighted to zero before they effect this regression and final solution. The default setting of TEST(61)=0 disables this test, as it has been largely superseded by the other consistency tests described above.

Bad sqrt argument

IASP91 software error message (subroutine FINDTT)

Failed to find phase XXXXXXXX

IASP91 software error message (subroutine FINDTT)

More than XXX arrivals found

IASPEI software error message (FINDTT) indicating that the number of arrivals in HYPPARM.INC (NIASP, default=150) has been exceeded. This usually indicates another problem with the IASP91 software, such as an unusual earth model.

Bad interpolation on XXXXX

Problem in IASP91 routine SPFIT. Again, it may be due to an unusual earth model.

Bad Tau: XXXX.XXX or Bad Range: XXXX.XXX IASP91 software errors in subroutine TAUINT

8 COMPUTATIONAL PROCEDURES

8.1 Program Flowchart

8.2 Adaptive Damping

The adaptive damping procedure is essentially the same as that described by Lienert et al (1986). The damping, which is a constant added to the diagonal terms of the G^TG partial derivative matrix product before inverting it to get the hypocentral corrections, is varied depending on whether the solution is converging or not. If the r.m.s. travel time residual increases, the minimum r.m.s. hypocenter is returned to, the damping is increased and the step is recalculated. A similar approach is used to prevent the depth from becoming negative. When the r.m.s. starts decreasing again, the damping is reduced, but at a slower rate than the initial increases. Previous programs such as HYPO71 solved the problem of instability in the solution by fixing the depth when the solution became unstable. Unstable solutions occur due to the geometry of the travel time paths, e.g., when the hypocenter is level with all the stations, or when all the stations lie on a straight line. With adaptive damping, the damping is varied while the iterations proceed, following the principle that the best solution is the one that best fits the data, i.e., it has the lowest r.m.s. residual. This technique often allows the hypocenter to move through "barriers of instability" that can occur, particularly with respect to hypocentral depth. The price that is paid is that the in the resulting solution, the parameters lose their independence, i.e., they become correlated with each other.

A well-known example is the correlation between the origin time and the depth of an event at a great distance from a small network.

Although the adaptive damping technique has proved very successful in general, cases have arisen where "multiple minima" occur in the r.m.s. residual. In this case, HYPOCENTER will simply stop at the first "r.m.s. hill" that it encounters by refusing to allow increases in the r.m.s. I have therefore made a small modification to the program which allows temporary increases in r.m.s. for a specified number of iterations (the default is 2). This allows trial steps to be taken up the "hill" to see if there is a "valley" on the other side, while retaining the option of returning to the minimum r.m.s. solution if necessary, i.e., if there is no subsequent decrease in r.m.s.

8.3 R.M.S. Calculation

A subprogram (RMSV) has been written to calculate the r.m.s. residual misfit of a set of travel times for a given hypocenter and origin time. This was essential, as both HYPOLOC and STARTLOC use RMSV to evaluate starting locations, the iterative solution and perturbations to the solution. RMSV does this by calling TRTIME, which calculates the theoretical travel times for each arrival and their partial derivatives using either the layered velocity model or the IASP91 routines. It then calculates the mean weighted residual, DTAV, of all the valid absolute travel times. DTAV is then subtracted from all the residuals, DT. An important feature of RMSV is that it does not adjust the origin time, TPH. Instead, it returns DTAV, which is then added to the origin time in HYPOLOC immediately before the next call to RMSV. In this way, the uncentered origin time is the value stored in TPH1, allowing the minimum value of r.m.s. residual to be correctly recalculated (the initial origin time can determine the IASP91 phases that are matched to the arrivals when the auto-picking feature, TEST(72)=1, is enabled). RMSV and TRTIME are described in detail in section 8.5.

8.4 Starting Location

The subroutine STARTLOC was first written to address the problem of locating distant events with a regional network. In situations where an event is many network widths away, least-squares iterations often fail to converge and an estimate of the starting location has proved essential. Locations determined using the first-arrival station as a starting point can also be seriously affected by a single large error in one of the arrival times, sometimes causing the event to be mislocated on the wrong side of an elongated network. A systematic evaluation of the best starting location forces an evaluation of such non-unique cases, which are better resolved initially rather than obtaining a non-unique solution which can sometimes appear to be a very good one. STARTLOC is described in detail in section 8.5.

8.5 Outlier Removal

It is well-known (e.g., *Jeffreys*, **1962**) that the least squares technique is sensitive to the presence of a single large residual, particularly when the number of data is small. Earthquake location data are quite frequently contaminated by misread minutes, sometimes even hours, particularly when the arrival times straddle a minute, hour or day boundary. Although residual weighting can be an effective method of weighting out such "outliers", it often fails to weight out the true outliers, instead weighting out valid arrivals. For this reason, we have included several consistency tests for the arrival time data to identify and weight out such outliers before they affect the solution.

The first check for outliers is that all valid (IPS=0, DTWT> 0) "S" and "L" phases do not precede their primary phase, "P" (only *S* and *P* first arrival phases are tested, as many of the more complicated phases at great distances, such as *SKP* and *PcP*, do not follow this rule).

The second check compares the travel times between pairs of identical phases at different stations sorted in arrival time order. The travel time difference for each phase pair is then compared to the theoretical travel time for that phase between the two stations, assuming that the event is at one of the stations. If the absolute difference between the arrival times and the theoretical travel time is less than TEST(60) seconds, the process is repeated with the next adjacent pair of phases, "leapfrogging" down the arrival time set. If the test fails, an error message is printed and the phase is weighted to zero. Its residual is therefore still calculated and can be checked in the printout.

Both these consistency tests can be disabled by setting TEST(60)=0 (not the default).

8.6 Residual Weighting

We weight the travel time residuals using *Andersons*, **1982** bisquare method, i.e., an absolute residual based taper, normalized to 4.685 times the median of the absolute residuals and starting at a minimum residual of 0.5 seconds. We found that a minor adaptation of this method resulted in a considerable improvement in its performance. The arrival time mean, on which the weights are based, can be biassed to one side by the presence of a single large positive or negative outlier causing a similar bias in the residual weights. We therefore recalculate the weighted mean value of the residuals a second time using the first set of residual weights and recalculate the weights a second time using this new mean. Since any severe outlier is weighted to zero the second time around, this second set of weights is unaffected by it.

8.7 The IASPEI Software

The IASP91 software package was originally written by Ray Buland (*Buland and Chapman*, **1983**). It generates travel times for most of the phases used in earthquake location. This phase set currently includes: *P*, *Pdiff*, *PKP*, *PKiKP*, *PP*, *PPdiff*, *PPKP*, *PPKiKP*, *sP*, *sPdiff*, *sPKP*, *sPKiKP*, *PP*, *P'P'*, *S*, *Sdiff*, *SKS*, *PS*, *PSdiff*, *PSKS*, *sS*, *sSdiff*, *sSKS*, *SS*, *S'S'*, *PS*, *PKS*, *SP*, *SKP*, *SKiKP*, *PcP*, *PcS*, *ScP*, *ScS*, *PKKP*, *PKKS*, *SKKP*, and *SKKS*. The IASPEI model was developed specifically as a replacement for the Jeffreys-Bullen model for earthquake location work. It incorporates a PEM core, a lower mantle developed by Ken Toy, and an upper mantle constructed by Brian Kennett. The phase set was chosen by requiring that the phases be well observed in some distance range and that they be useful either for earthquake location or for studies of earth structure.

For the Nordic phase data input format, which HYPOCENTER now reads as standard, a search is made for the IASP91 phase which corresponds to the 8-character phase identifier (columns 11-18). If no IASP91 phase is found, the phase is given a weight of -1, which effectively removes it from the phase set. However, if a phase is labelled as "P ", "S ", "PKP " or "SKS ", and this phase is not in the IASP91 list, the first arrival phase having "P" or "S" as its first letter is used, or "PKP ", "SKS " as its first 3 letters. In addition, "PKiK" phases are included in the search for "PKP " and "SKiK" phases in the search for "SKP ". Although such phases are rarely observed, they are continuous with *PKP* and *SKP* at 140 degrees, and provide continuity to the inversion process by allowing smooth movement into the *PKP* and *SKP* regions across the core-mantle boundary.

8.8 Program Module Description

HYPOCENTER was originally written to run on any standard 80x86 class of PC-compatible, which made the program compatible with a wide variety of platforms (SUN, VAX, etc.) having an ANSI-standard Fortran 77 compiler. However, with the addition of the global location capability, the MSDOS memory limit of 640 Kbytes became a serious obstacle. Initially, the IASP91 travel time software was modified to include an extra random-access file, reducing the memory requirements. However, this modification caused such a serious degradation in speed that it became impractical to locate distant events on-line. Also, the complete ISC station list consists of almost 5000 stations and some of the larger earthquakes result in over 1000 phases, which further increased the required size of the data arrays in the program.

I have therefore written this version of HYPOCENTER to run on an 80386 or better CPU using a memory extender. The hardware requirement is either an 80386DX or SX (with an 80387 coprocessor) or a 486DX (or SX with a coprocessor) or a Pentium, having at least 4 Mbytes of RAM. HYPOCENTER has been compiled on the MS Fortran Powerstation platform, which implements a flat (non-segmented) memory model using a memory extender which is compatible with (but does not require) MS-Windows extended memory. HYPOCENTER will execute under MS/PCDOS 3.3 or higher and requires the file DOSXMSF.EXE (included on the distribution disk) which implements the 386 memory extender. The amount of hard disk space required by the program and its associated files is now about 2 Mbytes. However, the listing file, PRINT.OUT can easily become several

Mbytes in size for large ISC input files and I would recommend having at least 10 Mb of free hard disk space available after the program is installed. HYPOCENTER has also been successfully compiled on SUN Work stations. On other platforms such as SUNS, the two random-access files IASP91.HED and IASP91.TBL need to be generated using the programs REMODL and SETBRN, as these two files are binary and therefore have machine-dependent formats. I have also put the sizes of the large data arrays in an include file, HYPPARM.INC, so as to make increasing their size a straightforward matter.

Below is a list of all files distributed with an indication of differences between Sun and PC.

All names with an ".FOR" extension are source code, ".INC" are include files and those with ".EXE" are binary executables .

Module Name	Comments
HYP.FOR	main program
HYPOSUB1.FOR	HYPOCENTER source
HYPOSUB2.FOR	" "
HYPOSUB3.FOR	use F77 -NL2000 on SUN
HYPOSUB4.FOR	" "
HYPOSUB5.FOR	used with SEISAN
TAU.FOR	IASP91 software
HYP.EXE	executable
DOSXMSG.EXE	Phar-Lap DOS memory extender (PC)
IASP91.HED	binary random-access file, different on Sun and PC
IASP91.TBL	" "
STATION0.HYP	test parameters, station data, velocity model
COMPRESS.EXE	remove phase data from output files
ISCNOR.EXE	convert ISC phase data to NORDIC
HYPNOR.EXE	convert HYPO71/HYPOINVERSE phase data to Nordic
TTIM.EXE	calculate selected IASP91 travel times

8.8.1 MAIN PROGRAM (HYP.FOR)

The main calling routine performs residual the following tasks:

1. Reads the name of the input file (NORDIC format).

2. Opens the input and output files.

3. Performs initialization calls to HYPOCENT (with IFUNCT=1 and IFUNCT=4).

The program either runs interactively on an event by event basis, or processes all the data in the specified input file, depending on whether interactive mode is requested. For each event, it performs a location call to HYPOCENT (IFUNCT=2), writes the results into the output files and prints the location and residuals to the screen in interactive mode. Finally, when "Q" is selected, it performs a call to HYPOCENT (IFUNCT=3) which calculates and outputs a residual summary for all of the selected events.

8.8.2 HYPOCENT (in HYPOSUB1.FOR)

HYPOCENT is the driver for the location software. Its function depends on the value of the first parameter IFUNCT in its input window. The large data arrays for the station coordinates, phase ID's and times, etc., are dimensioned at the start of HYPOCENT. Except for a common block containing the partial derivative matrix (defined in the include file COMM1.INC), these arrays are then passed as arguments to the subroutine HYPOLOC, which does the location.

HYPOCENT is called for the following tasks:

1. Initialization (IFUNCT=1):

   (a) Sets the TEST parameter defaults by calling subroutine SETTEST

   (b) Performs the initialization calls to the IASP91 software by calling TABIN and BRNSET.

   (c) Reads the IASP91 file IASP91.HED, station data (calling STATIONS), finds the maximum elevation station, then reads in the layered velocity model, trial depth and V_p/V_s ratio. If the V_s velocities are zero, it uses the V_p/V_s ratio to calculate V_s for each layer. It then calculates the layer thicknesses. If TEST(40)=0, specifying sea level as the depth origin, it adds the maximum station elevation to the thickness of the top layer. For local seismic networks, it is better to set TEST(40)=1, which then uses the maximum elevation station as the depth origin. This allows the velocities to be specified all the way up to the maximum elevation station. The program was specifically modified to allow for the case where events could occur above the stations locating them. However, in this case, the layered model is assumed to be present at all points between the station and the event location.

2. Location (IFUNCT=2):

   (a) Reads a set of input phases by calling GETPHASN. GETPHASN converts the arrival times to seconds after 1/1/1900, using the day, month and year on the preceding type 1 record and stores these times in the double precision array TP. If either a "B" or "N" layer is specified in the velocity model, stores the layer indices in NCONRAD and NMOHO, respectively.

   (b) Determines whether depth, epicenter or origin time are to be fixed.

   (c) Converts any starting location information to geocentric radians on a sphere of radius 6378.2 km (the earth's equatorial radius) using subroutine FOLD. The internal coordinate system used by HYPOCENTER takes east longitude, north latitude and depth down as positive.

   (d) Matches each phase's station with one in the station list, converts the station latitude and longitude to geocentric radians and stores this with other station parameters in separate arrays with indices matched to the phase index. If azimuths are read as input, an additional phase record is generated for them.

   (e) Calculates and stores the phase weights in the array DTWT.

   (f) Performs a call to HYPOLOC to locate the event.

   (g) Calculates additional information about the hypocenter (azimuthal gap, distance, residual information, etc.) and writes this information into the file PRINT.OUT. The residuals are also accumulated in separate arrays for each station for the residual summary.

3. Residual Summary (IFUNCT=3):

   This call prints the mean residuals, their number and r.m.s. deviation for each station in the list for all of the located events. This listing is useful for determining systematic station delays which can then be corrected for by entering the delay as input with the station list. A mean r.m.s. residual value for all the located events is also determined.

This "overall misfit" is useful for determining whether changes in the velocity model, for example, have any systematic effect on the results for a large number of events.

4. Zero Accumulated Residuals (IFUNCT=4):

   This call was separated from the initialization call so as to allow a new set of residual results to be started at any point. However, this has not yet been implemented in the calling program.

8.8.3 HYPOLOC (in HYPOLOC.FOR)

HYPOLOC contains the iteration loop within which the travel time and matrix inversion routines are called to obtain a hypocenter solution. Its major tasks are described below:

1. Checks for the following non-locatable or ambiguous cases, provided that the hypocenter is not fixed. HYPOLOC exits with an error message and ICD=6 in these cases.

   (a) Less than 3 valid phases and no azimuths

   (b) Less than 2 valid phases and 1 azimuth

   (c) Valid phases at less than 3 different stations and no azimuths

   The last case is one that other programs will give a solution for, often without any indication that there are two equally valid solutions, e.g., in the case of multiple arrivals at two stations. I concluded that it is better to exclude this case, forcing an operator to enter an azimuth, or data from another station, rather than to output an ambiguous solution.

2. Sorts the arrival time indices in order from first to last in the array ISORT. The first arrival then has the index ISORT(1), the second ISORT(2), etc., in the original array, which is left unchanged.

3. Looks for difference times (e.g., *P-S*) with the primary phase weighted to 9 and sets the array IPS to 1 for all their secondary phases. Phases are considered to be secondary if their first letter does not match the first letter of the phase weighted to 9.

4. Provided TEST(60) is not zero, performs a consistency check on the arrival times.

   The consistency check was introduced as a way of removing arrivals that are obviously bad (e.g., off by minutes or hours) before they affect the starting location and solution.

5. Calls the starting location subroutine STARTLOC. The starting depth returned by STARTLOC is limited to TEST(70) at this point.

6. Starts the iteration loop with the depth initially fixed. The parameter change limit, TEST(32), is initially increased by a factor of 10. The following calculations are made during each iteration:

   (a) The r.m.s. residual, RMS (σ), is calculated (using RMSV) for the current hypocenter. For the first iteration, σ is stored in RMS1, the minimum r.m.s. residual (σ_0), forcing the calculation of an inverse matrix on the first iteration. The residual average (DTAV) is added to the origin time (TPH) before calling RMSV. This means that the minimum r.m.s. residual origin time, TPH1, is not corrected for the mean residual, DTAV, calculated by RMSV.

   (b) If $\sigma \leq \sigma_0$, HYPOLOC calculates an inverse matrix and determines its eigenvectors and eigenvalues using MINV. It also stores the current set of residuals, DT in DT1, RMS in RMS1 and the current hypocenter, XH in XH1. The parameter corrections, DTH, are then calculated using CORR with the current damping factor, XK. Provided DTH(3) would not make the depth negative, the parameter corrections are applied to XH. The magnitude of the parameter change, DTH, is then calculated to test for convergence. The eigenvector and eigenvalue arrays, V and EIG, as well as the residual array, DT1, are therefore only updated when $\sigma \leq \sigma_0$ and are always calculated at the minimum r.m.s. residual solution, XH1. After the first iteration, the damping is reduced by a factor of 4, or 1.5 if σ has previously increased. However, it is not allowed to go below its specified starting value, TEST(30).

   (c) If $\sigma > \sigma_0$, or if a negative depth is requested, XH is returned to the minimum r.m.s. residual value, XH1, the damping factor XK is increased by the factor TEST(39), and the parameter corrections are recalculated. If the number of increases in XK exceeds TEST(37), the depth is fixed and XK is returned to its initial value. Increases in σ or negative depths are not counted as an iteration, as the repeat corrections with increases in damping are done within the iteration loop.

7. If the iterations converge, or the maximum number of iterations, TEST(11), is exceeded, HYPOLOC calculates the errors in the minimum r.m.s. residual solution,

the resolution matrix and converts the latitude and longitude back to geographic coordinates using UNFOLD. The minimum r.m.s. residual origin time, TPH1, is also corrected by adding the mean residual, DTAV, to it.

8. If TEST(13)>0, HYPOLOC calculates the changes in the r.m.s. residual at 6 points TEST(13) km either side of the solution. This provides a useful check on whether the minimum r.m.s. residual has actually been reached. All the DRMS values should remain positive if this is the case.

8.8.4 STARTLOC (in HYPOSUB4.FOR)

STARTLOC takes as input a set of travel times, which must be matched to their station IDs and coordinates (geocentric) and must each have a minimum of 4 characters in a phase identifier ("PN ", "PG ", "P ", "SN ", "SG ", "S ", "L ", "AZ ", "PKP ", etc.) where a blank second letter means an unidentified first arrival. The arrival times/azimuths should be in the double precision input array TP. Times must be seconds referenced to a fixed datum (e.g., seconds from 1/1/1900), while azimuths are in degrees, clockwise from North, measured relative to the receiver. Station latitude and longitude are in geocentric radians. The r.m.s. residual misfit of all candidate starting locations is calculated using RMSV and the minimum r.m.s. residual (RMS1) is then returned as the starting location.

The starting location strategy proceeds as follows:

1. Look for depth phases prefixed by "p". These are then used to obtain an approximate starting depth using the difference in time between the depth phase and its corresponding first arrival (e.g., pP_n and P_n). I then use the velocity defined by TEST(69) to calculate the depth (the default value is 5 km/s). Although the apparent velocity calculated from pP-P differences in the table given in Richter (1970) varies between 3.3 and 6 km/s, the value of 5 appears to give a good enough starting depth for the program itself to obtain a better depth using the pP partial derivatives. In order to exclude severe outliers, I calculate an r.m.s. residual of all the depths (when there are 3 or more) and exclude depth values of greater than twice the r.m.s. residual.

2. Calculate a minimum value of the r.m.s. residual (using RMSV) from the position 0.15 km N and E of the first-arrival station. Provided the first-arrival is not an "N" phase, I set RMS1 to RMS. If TEST(56)<0, exit at this point. For all of the following starting locations, set RMS1=RMS if RMS is smaller.

3. Use either the starting location depth given as input to STARTLOC, or the value from (1) as the depth for all subsequent starting locations.

4. Determine MAXNO, the index of the (absolute) phase recorded at the maximum number of stations . If MAXNO>2 perform a linear regression on the arrival times with respect to their latitudes and longitudes to determine their azimuth, θ_a and apparent velocity, V_a. To test for outliers in the regression, I estimate the r.m.s. residual of the deviations in the arrival times from those predicted by the regression. If any deviation exceeds TEST(61) times the r.m.s. residual, the arrival time is rejected and the regression repeated without it. I then estimate the distance, Δ (degrees), to the event from the center of the network using the formula

$$\Delta = (10.46 - 110.7/V_a)/0.067 \tag{27}$$

These coefficients were determined using a least-squares linear fit to data obtained from the Norwegian seismic network, using over 100 earthquakes and ISC locations of the same events, 0 to 170 degrees from the network center. Although the theoretical variation of slowness versus distance is not linear, I found that the observed deviations from the theoretical curves were sufficient to justify a simple linear fit. To further improve the starting location, I have implemented a simple "grid-search" type algorithm which increases or decreases Δ and θ_a at 1 degree intervals, looking for a minimum r.m.s. residual, using the routine RMSV. This technique has proved so effective that very few iterations are needed on the initial starting location in order to obtain a solution. However, it has occasionally found spurious minima at large distances for local events. To prevent this, I only perform the minimum search if the first r.m.s. residual is less than RMSMIN or if the distance indicator, DISTIND='D'.

5. Determine the number of azimuthal phases, NAZ. Exit if MAXNO=0 and NAZ=0 (not locatable). Use the azimuth data to define starting locations based on intersections with other azimuths, including AZAPP, by calling AZCROS.

6. Find NMULT, the number of multi-phase (S-P, etc.) stations and sort them in arrival time order. Calculate the distances of all of these phases from their respective stations using the moho velocity for "N" and blank (minimum) phases and the layer 1 velocity for "G" phases. This section has not been written to use the IASP91 software and

may give unreliable results at large distances, particularly with unusual phases (*SKP* etc.).

For all these calculated distances, I then calculate potential starting locations for all of the following cases:

(a) An azimuth at the same multi-phase station

(b) The intersection of the apparent velocity azimuth, AZAPP, with each distance

(c) The intersections of pairs of distances at two different stations. Since in this cases, I get two locations for each distance pair, I test all valid intersections, taking the one with the minimum r.m.s. residual

(d) An azimuth at a different station

7. Search for any refracted ("N") and core ("K") phases at stations having an azimuth and test the location defined by the critical distances, TEST(53) and TEST(59), respectively, and the azimuth. If there are no azimuths, use the apparent velocity one and the distance from the center of the network.

8.8.5 RMSV (in HYPOLOC1.FOR)

RMSV calculates the r.m.s. travel time residual of a hypocenter with respect to a network of stations using the following steps:

1. The distances, DELTA from the current hypocenter are calculated for all phases. If any of them exceed TEST(57) and DISTIND is not set to local ("L"), IFLAGD is set to 1, which forces the use of the IASP91 software in TRTIME. Also, if the distance indicator is set to "D", IFLAGD is set to 1. Otherwise (including "R" events), IFLAGD=0, which forces TRTIME to use the layered model.

2. TRTIME is then called for all phases.

3. The difference phase (IPS(I)=1) time differences, their derivatives and weights are calculated.

4. The residuals, DT(I) and the weighted mean of all the absolute times, DTAV, are calculated. RMSV, is then calculated and DTAV is subtracted from all the absolute DT(I)'s.

5. If TEST(36)>0, the residual weights are calculated. Steps (ii) to (iv) are then repeated using the residual weights.

8.8.6 TRTIME (in HYPOLOC1.FOR)

TRTIME was written to provide a consistent interface to the two types of travel time calculations, the layered model (DTDX2) and the IASP91 software (TRTM). It also determines how to match the requested (i.e., operator defined) phases with those that are possible at the position of the current hypocenter. TRTIME performs the following:

1. The input variable MINFLAG=TEST(63), is used to determine the types of input phases (stored in the PHASE array) accepted, based on their second letter (converted to upper case).. For any phases that are not matched as described below, TRTIME returns with an error message and DTW1 set to -1. Otherwise, the CHARACTER*1 variable PRMD is assigned one of the values described below.

   MINFLAG=0: PRMD=PHASE(2:2) (second letter of phase).

   MINFLAG=1: PRMD=" ".

   MINFLAG=2: PRMD= "N".

   MINFLAG=3: PRMD="G".

   MINFLAG=4: PRMD="B".

   The distance in degrees (Δ) and azimuth (AZZ) of the event from the phase's station is now calculated using DISTAZ.

2. *P* and *S* phases: If the first letter of the phase (converted to upper case) is "P" or "S", TRTIME proceeds to calculate a travel time using either DTDX2 or IASP91.

   (a) Layered model times (IFLAGD=0 or DISTIND="L"):

   If the first phase letter is not blank and PRMD is "N", "G" or "B", DTDX2 is used to calculate the travel time and partial derivatives of the phase. Before calling DTDX2, I add the maximum elevation to the hypocentral depth if TEST(40)=0 (depth origin at sea level). DTDX2 always uses the maximum elevation station as its origin, regardless of the setting of TEST(40). The station "depths" are likewise calculated by subtracting each station's elevation from the maximum elevation station. If DTDX2 returns with IFLAG=0, signifying an invalid phase, I change PRMD from "N" to "B", then from "B" to "G".

This prevents phases being weighted out close to the first-arrival station which artificially lowers the r.m.s. residual and can cause problems with the starting location in STARTLOC. If DTDX2 still returns with IFLAG=0, I set DTW1=-1 and return with an error message, which is printed to the screen if TEST(66)$\neq$ 0.

(b) IASP91 Times (IFLAGD=1 or DISTIND="D"):

The first step is to call DEPSET, the IASP91 routine which initializes for a new depth. To save calculation time, I store the previous depth in OLDDEP, and only call DEPSET if the depth has changed. TRTM is then called with the hypocentral distance in degrees, DEDEG (Δ), as input. TRTM outputs the number of phases (NIASP), and arrays containing the travel times (TT) , the phase ID's (PHCD) and the partial derivatives $\partial t/\partial\Delta$ (DTDL), $\partial t/\partial z$ (DTDH), all sorted in ascending order of the travel times (the units of the partial derivatives, DTDL and DTDH are both in seconds/degree). If test(72)=0 or IMATCH=1, an attempt is made to match the operator-specified phase ID, PHASE, with one of the PHCD phases. This has proved to be a rather complicated task, as conventions for naming global phases vary widely. However, I have done the best I can to match phases from a number of trial data sets from the CD-ROM of ISC data, obtained using the NEIC CD-ROM reader program, FAISE.

In order to obtain a match, I use a set of indices to look for the first arrival of a variety of phases which presently include *P*, *S*, *PP*, *PS*, *PKP*, *PKS*, *SKP*, *SKS*. I included logic to exclude each selected phase from selection by another index. Although this part of the program works reasonably well, the amount of effort required to handle all the possible permutations of possible phases is enormous. The major reason that I shall not devote the rest of my life to this problem (which I easily could) is that my experience with the ISC data set has indicated that many, if not most, of the assigned phase ID's other than "P" are of questionable validity. In order to identify more complicated phases, an operator must have some idea of the distance of the event being located. Although *S-P* differences and frequency content can give some indication of distance, such a distance determination should be the job of the location program. Operator-assigned phase ID's are in many cases inaccurate, subjective and misleading. I would prefer to have operators give the times, and ONLY the times, of all the clear first-breaks at a station without any making any attempt to identify them.

I have attempted to solve the problem of wrongly assigned phase ID's adding a parameter, TEST(72), which when set to 1.0 (not the default) allows TRTIME to iterate TEST(73) times while attempting to match the input phase ID's. After TEST(73) iterations, it matches all the observed arrivals (including blank ID's) with the closest calculated times in the complete IASP91 set. Results so far using this technique have been encouraging. Using this technique, HYPOCENTER frequently converges on a solution for large numbers (>500) of arrivals without any residual weighting. Even if it is not routinely used for location, this option should prove useful in identifying the closest IASP91 arrivals automatically once the event is located. This can be done by specifying a fixed hypocenter while setting test(73)=0 and test(72)=1.

Having found an IASP91 travel time, I determine the station elevation correction, as the IASP91 program performs all its locations on the a sphere having the earth's mean radius (6371 km), i.e., its depth origin is at sea level. The velocities that I use for the elevation correction are the first layer velocities (V_p and V_s) in the layered model (in STATIONX.HYP). I then calculate the vertical slowness

$$p_z^2 = \frac{1}{V^2} - \left(\frac{\partial t}{\partial \Delta}\right)^2 \tag{28}$$

where $\partial t/\Delta$ is the IASP91 derivative, (i.e., the horizontal slowness in sec/deg.) converted to sec/km and V is the surface velocity in the IASP91 model. The correction which is added to the calculated travel time, is then $t_c = p_z z_e$, where z_e is the stations elevation above sea level in km. To determine the angle of incidence at the source, ϕ_i, it is necessary to determine the velocity, V^*, at the hypocenter depth. This is obtained using the IASP91 subroutine EMIASK, which is used by his program REMODL to define the IASP91 earth model P and S velocities. ϕ_i is then given by

$$\phi_i = \sin^{-1} V^* \frac{\partial t}{\partial \Delta} \tag{29}$$

3. L and R phases: TRTIME evaluates both these times and their derivatives using the constant velocities defined by V_R=TEST(44) and V_L=TEST(51). The derivatives are then $-1/V_L$ and $-1/V_R$, resolved into the north and east directions using the event's azimuth, θ_e and converted to seconds/geocentric radian.

4. Azimuthal derivatives: These are calculated using the two equations

$$\frac{\partial \theta_e}{\partial x} = \frac{-\cos\theta_b}{\sin|\Delta|} \tag{30}$$

and

$$\frac{\partial \theta_e}{\partial y} = \frac{-\sin\theta_b}{\sin|\Delta|} \tag{31}$$

where θ_b and Δ are the back-azimuth and distance of the epicenter obtained using DISTAZ. In order to keep the azimuth residuals from having an unduly large weight compared to the time residuals, I divide the azimuth residuals by an "azimuth error" which is specified by the parameter TEST(52). The default value of 10 degrees makes an azimuth residual of 10 degrees equivalent to a 1 second residual in arrival time. The azimuth residuals (and difference phases) are excluded from the calculation of the mean residual, DTAV, which is used to determine the origin time.

5. Azimuth residuals: When calculating these, it is necessary to consider the cases where the azimuths change sign, such as close to 180 degrees. Otherwise, a residuals of 359 degrees can result when subtracting -180 from 179, for example. I first ensure that both the observed and calculated azimuths are >0 by adding 360 to them if they are <0. I then calculate the difference, adding 360 to residuals < -180 and subtracting 360 from those > 180. Distance weighting is then added to the calculated azimuth weights, DTWT, to obtain the final output weight, DTW1.

8.8.7 DTDX2 (in HYPOSUB2.FOR)

DTDX2 was written to calculate travel times and their derivatives for a simple layered velocity model. Either the receiver or source can be at any point below defined depth origin, which is always at the top of the first layer. This was specifically written for a rather unusual situation in Hawaii, where ocean bottom seismometers were locating earthquakes above sea level, on the island of Hawaii. The program basically follows the method described in Lee et al. (1981). However, it has been adapted to allow the forcing of "n", "b" and "g" phase calculations. The major steps are described below:

1. Get delta and azimuth of the event using DELAZ. Calculate difference in longitude and latitude and distance between station and epicenter. Check for trivial cases, e.g., hypocenter below moho for "n" phase.

2. Find the travel times for all possible refracted waves and their critical distances. Store the minimum valid travel time in TMIN.

3. Find the direct travel time and set TMIN to it if it is less than the minimum refracted time.

4. Calculate partial derivatives and the angle of incidence for the minimum travel time and exit.

8.8.8 ELLIPCOR (in HYPOLOC1.FOR)

This subroutine interpolates the tables of ellipticity corrections given by *Dziewonski and Gilbert*, **1976**, for the phases *P, PcP, PKPab, PKPbc, PKiKP, S, ScS, SKS* at hypocentral depths of 0, 300 and 650 km. The interpolation is done using polynomial fits to each of the tables. A polynomial degree of 8 was empirically determined to give an accuracy of better than 1% to all the values in the tables. Since the maximum ellipticity correction is -0.75 seconds at a distance of 105 degrees, this accuracy is more than adequate. Ellipticity corrections of zero are returned for phases other than those listed above.

8.8.9 TTCHECK (in HYPOLOC.FOR)

The purpose of this subroutine is to remove arrivals that are obviously bad (e.g., off by minutes or hours) before they affect the starting location and solution. The first consistency check tests that all valid (IPS=0, DTWT>0) "S " and "L" phases do not precede the primary phase, "P " (only S and P first arrival phases are tested, as many of the more complicated phases at great distances, such as *SKP* and *PcP*, do not follow this rule).

The second check compares the travel times between pairs of identical phases at adjacent stations. The travel time difference for each phase pair is then compared to the theoretical travel time for that phase between the two stations, which is calculated using the layered (local) velocity model. If the absolute difference between the arrival times and the theoretical travel time is greater than TEST(60) seconds, the test fails. I then attempt to find which of the two arrivals is bad by repeating the check a second time with the next closest station to each of the two stations. If either of the second checks fail, that station's arrival is weighted to zero. However, if both of the second checks succeed, I am forced to weight both arrivals to zero.

8.9 Compilation and Linking:

Recompiling HYPOCENTER from scratch (e.g., on a different computer) involves two steps.

1. Generating the two direct-access files, IASP91.HED and IASP91.TBL. To do this, compile and link REMODL.FOR, TAU.FOR and EMIASP91.FOR, then run REMODL.EXE (no input is required). Then compile and link SETBRN.FOR and TAU.FOR (you need a 32-bit compiler for SETBRN) and run SETBRN. This will generate the two files IASP91.HED and IASP91.TBL in a format compatible with your computer. Put these two files in the same directory as the one in which you generate the HYPOCENTER executable, HYP.EXE

2. Generate HYP.EXE using the following modules: For convenience, I have listed the subroutines in each source file. If you are using Fortran Powerstation or a "MAKE" utility, add all these source files to the project.

Module name	Subroutines
HYPOCENT.FOR	MAIN, HYP_PRINT, FIND_FILE_TYPE
HYPOSUB1.FOR	HYPOCENT, FOLD, BACK, CVRTOP, HYPOT, DELAZ, UNFOLD, SETTEST, GETPHASE, STATIONS, UCASE, AZCROS, DISTAZ, CART, GEOG, ROTATE
HYPOSUB2.FOR	DTDX2, AZDIST, MINV, CORR, MEDIAN, GETPHASN, DEL, EIGEN, FDIST, QF, BETAI, BETACF, GAMMALN
HYPOSUB3.FOR	UPDATE_MAG, READ_STAT_MODEL,
HYPOSUB4.FOR	STARTLOC, LUDCMP, LUBKSB, LCASE
HYPOSUB5.FOR	TOPDIR, GET_AGENCY, COMPUTER_TYPE, DIR_CHAR, SYS_CALL, NUMBER_ARGS, GET_ARG, SET_CBREAK, UNSET_CBREAK, SYSTEM_C, INDATA, FIND_EV_INDEX, TOPDIR, ABSTIM, TIMSEC, SECTIM, DTE, TIMADD, MONTH_DAY, DATE_DOY NB: HYPOSUB5.FOR is not present if HYPOCENTER is part of the SEISAN system,since all these routines are part of SEISAN
HYPOLOC.FOR	HYPOLOC, TTCHECK
HYPOLOC1.FOR	RMSV, TRTIME, R8SORT, EMIASK, ELLIPCOR
TAU.FOR	ASNAG1, ASSIGN, BKIN, BRNSET, DEPCOR, DEPSET, FINDTT, FITSPL, IUPCOR, PDECU, QUERY, R4SORT, RETRNS, TABIN, TAUINT, TAUSPL, TRTM, UCTOLC, UMOD, ABORT, DASIGN, GETARG, GETCL, IARGC, NXARG, TNOUA, VEXIT
HYPPARM.INC	include file containing array dimensions
COMM1.INC	include file containing common block

9 REFERENCES

Anderson, K.R., (1982). Robust earthquake location using *M*-estimates, *Phys. Earth Plan. Inter.*, **30**, 119-130.

Buland, R. and C.H. Chapman (1983). The computation of seismic travel times, *Bull. Seism. Soc. Amer.*, **73**, 1271-1302.

Dziewonski, A.M. and F. Gilbert (1976). The effect of small, aspherical perturbations on travel times and a re-examination of the corrections for ellipticity, *Geophys. J. Roy.*

Astron. Soc., **44**, 7-17

Eaton, J.P. (1969). HYPOLAYR, a computer program for determining hypocenters of local earthquakes in an earth consisting of uniform flat layers over a half space, *U.S. Geol. Surv. Open-file Reort*,

Evernden, J.F. (1969). Precision of epicenters obtained by small numbers of world-wide stations, Bull. Seism. Soc. Amer., **59**, 1365-1398.

Flinn, E.A. (1965). Confidence regions and error determinations for seismic event location, *Rev. Geophys.*, **3**, 157-185.

Havskov, J. (1990). Replace the HYPO71 format?, Bull. Seism. Soc. Amer., **80**, 1043-1045.

Havskov, J. and T. Utheim (1992). SEISLOG and SEISAN: A complete system for seismic data acquisition and analysis. *Cahier du Centre Europeen de Geodynamique et de Seismologie*, **5**, 67-74.

Havskov, J. and Lindholm, C. (1994). The SEISAN earthquake analysis software for the IBM PC and Sun, version 4.0. *Manual, Institute of Solid Earth Physics, University of Bergen, Norway*, 139 pp.

Hutton, L.K. and D.M. Boore (1987). The M_L scale in Southern California. *Bull. Seism. Soc. Amer.*, **77**, 2074-2094.

Jackson, D.D. (1972). Interpretation of inaccurate, insufficient and inconsistent data, *Geophs. J. Roy. Astr. Soc.*, **28**, 97-109.

Jeffreys, H. (1962). *The Earth*, Cambridge Univ. Press, Oxford.

Jordan, T.H. and K.A. Sverdrup (1981). Teleseismic location techniques and their application to earthquake clusters in the South-Central Pacific, *Bull. Seism. Soc. Amer.*, **71**, 1105-1130.

Kennett, B.L.N. and E.R. Engdahl (1991). Traveltimes for global earthquake location and phase identification, *Geophys. J. Int.*, **105**, 429-465.

Lahr, J..C. (1978). HYPOELLIPSE: A computer program for determining local earthquake hypocentral parameters, magnitude, and first motion pattern. *USGS Open File Report, USGS, Menlo Park CA*, 34 pp

Lee, W.H.K. and J.C. Lahr (1978). HYPO71 (revised): A computer program for determining hypocenter, magnitude, and first motion pattern of local earthquakes, *USGS Open File Report 75-311*, 448p.

Lienert, B. R., E. Berg and L. N. Frazer (1988). HYPOCENTER: An earthquake location method using centered, scaled, and adaptively least squares. *Bull. Seism. Soc. Am.*, **76**, 771-783.

Minster, J.B., T.H. Jordan, P. Molnar and E. Haines (1974). Numerical modeling of instantaneous plate tectonics, *Geophys. J.*, **36**, 541-576.

Veith, K.F. and G.E. Clawson (1972). Magnitude from short-period *P*-wave data. *Bull. Seism. Soc. Amer.*, **62**, 435-440.

10 APPENDIX I: Additional Programs

All of the following programs are stand alone modules which can be compiled using the SUN F77 compiler, Microsoft Fortran 5.x, Powerstation, etc.

ISCNOR

This program converts data from the NEIC Associated data CD-ROM to Nordic Format, suitable for input to HYPOCENTER. It also takes the ISC location, errors, r.m.s. residual, etc. and writes them into the phase header line. The source module for this program is ISCNOR.FOR. It is run by typing "ISCNOR". It will then prompt the user for an input and output file. The output file can then be read by HYPOCENTER and relocated with no modification.

HYPNOR

This program converts HYPO71 phase format to Nordic. Only one letter of phase ID is output, as this is the maximum length of the HYPO71 phase ID. Since the station and phase data are combined in one file in the HYPO71 format, the phase data must be separated before running HYPNOR on it. The station data can be put in the STATION0.HYP parameter file. The source module for this program is HYPNOR.FOR. Like ISCNOR, it only requires specification of an input and output file.

HINNOR

This program converts HYPOINVERSE phase format to Nordic. The source module is HINNOR.FOR. Again, it only requires specification of an input and output file.

COMPRESS

This program removes everything except the phase header lines from a Nordic-format phase data file. It is useful for compressing the HYP.OUT files generated by HYPOCENTER before using them as input to plot programs, etc. It is run by typing "COMPRESS". It will then prompt the user for an input and output file.

11 APPENDIX II: The IASP91 Software

In order to generate the two IASP91 files IASP91.HED and IASP91.TBL for either the IASP91 earth model or for a different earth model, Buland's program REMODL must be run. This program generates the IASP91 earth model to calculate the travel times, using the model defined in the subroutine EMIASP91 (in EMIASP91.FOR). You can modify the earth model by changing this subroutine. Although the default in HYPOCENTER is to call the model name "IASP91", I would encourage people to change this name to one of their own for different earth models. We have made it relatively straightforward to change the model name in HYPOCENTER using a single variable (modnam) in the subroutine READ_STAT_MODEL in HYPOSUB3.FOR (the default subdirectory can also be changed here). REMODL, once compiled, needs to be linked to two additional compiled modules, (TAU.OBJ and EMIASP91.OBJ on a PC or TAU.O and EMIASP91.O on a SUN). Once compiled and linked, it needs no input or output, simply run it by typing REMODL. It generates two random-access files, REMODL.HED and REMODL.TBL. These are read by the next program, SETBRN, which generates the two files used by HYPOCENTER. SETBRN.FOR is also linked with the compiled TAU.FOR module (TAU.OBJ on a PC). SETBRN again needs no input values, but will spit out a lot of stuff on the screen while running. This is normal. When you have run SETBRN, you can test it by compiling the main program TTIM.FOR and linking it with TAU.OBJ. TTIM calculates the travel times for selected phases at different values of delta and depth and is a useful program to have on its own.