UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

User's Guide to HYPOINVERSE-2000, a Fortran Program to Solve for Earthquake Locations and Magnitudes

4/2002 version

by
Fred W. Klein

U. S. Geological Survey
345 Middlefield Rd., MS #977
Menlo Park CA 94025
klein@usgs.gov

Open File Report 02-171
Version 1.0

4/2002 program version

TABLE OF CONTENTS

TABLE OF FIGURES AND TABLES

INTRODUCTION

Hypoinverse is a computer program that processes files of seismic station data for an earthquake (like p wave arrival times and seismogram amplitudes and durations) into earthquake locations and magnitudes. It is one of a long line of similar USGS programs including HYPOLAYR (Eaton, 1969), HYPO71 (Lee and Lahr, 1972), and HYPOELLIPSE (Lahr, 1980).

If you are new to Hypoinverse, you may want to start by glancing at the section "SOME SIMPLE COMMAND SEQUENCES" to get a feel of some simpler sessions. This document is essentially an advanced user's guide, and reading it sequentially will probably plow the reader into more detail than he/she needs. Every user must have a crust model, station list and phase data input files, and glancing at these sections is a good place to begin. The program has many options because it has grown over the years to meet the needs of one the largest seismic networks in the world, but small networks with just a few stations do use the program and can ignore most of the options and commands.

History and availability. Hypoinverse was originally written for the Eclipse minicomputer in 1978 (Klein, 1978). A revised version for VAX and Pro-350 computers (Klein, 1985) was later expanded to include multiple crustal models and other capabilities (Klein, 1989). This current report documents the expanded Y2000 version and it supercedes the earlier documents. It serves as a detailed user's guide to the current version running on unix and VAX-alpha computers, and to the version supplied with the Earthworm earthquake digitizing system. Fortran-77 source code (Sun and VAX compatible) and copies of this documentation is available via anonymous ftp from computers in Menlo Park. At present, the computer is swave.wr.usgs.gov and the directory is /ftp/pub/outgoing/klein/hyp2000. If you are running Hypoinverse on one of the Menlo Park EHZ or NCSN unix computers, the executable currently is ~klein/hyp2000/hyp2000.

New features. The Y2000 version of Hypoinverse includes all of the previous capabilities, but adds Y2000 formats to those defined earlier. In most cases, the new formats add 2 digits to the year field to accommodate the century. Other fields are sometimes rearranged or expanded to accommodate a better field order. The Y2000 formats are invoked with the "200" command. When the Y2000 flag is turned on, all files are read and written in the new format and there is no mixing of format types in a single run. Some formats without a date field, like station files, have not changed. A separate program called 2000CONV has been written to convert old formats to new.

Other new features, like expanded station names, calculating amplitude magnitudes from a variety of digital seismometers, station history files, interactive earthquake processing, and locations from CUSP (Caltech USGS Seismic Processing) binary files have been added.

General features. Hypoinverse will locate any number of events in an input file, which can be in one of several different formats. Any or all of printout, summary or archive output may be produced.

Hypoinverse is driven by user commands. The various commands define input and output files, set adjustable parameters, and solve for locations of a file of earthquake data using the parameters and files currently set. It is both interactive and "batch" in that commands may be executed either from the keyboard or from a file. You execute the commands in a file by typing

@filename at the Hypoinverse prompt. Users may either supply parameters on the command line, or omit them and are prompted interactively. The current parameter values are displayed and may be taken as defaults by pressing just the RETURN key after the prompt. This makes the program very easy to use, providing you can remember the names of the commands. Combining commands with and without their required parameters into a command file permits a variety of customized procedures such as automatic input of crustal model and station data, but prompting for a different phase file each time.

All commands are 3 letters long and most require one or more parameters or file names. If they appear on a line with a command, character strings such as filenames must be enclosed in apostrophes (single quotes). Appendix 1 gives this and other free-format rules for supplying parameters, which are parsed in fortran. When several parameters are required following a command, any of them may be omitted by replacing them with null fields (see appendix 1). A null field leaves that parameter unchanged from its current or default value. When you start HYPOINVERSE, default values are in effect for all parameters except file names.

Format of this document. Throughout this document, the file formats are given where appropriate.   Data files are fixed format (column oriented). The formats in this document use the fortran specification. Thus A10 is an alpha field 10 characters long, I5 is a five digit right-justified integer, F4.2 is a 4 column real number where the decimal point implied as 2 decimal places if the decimal point is not present, etc. Formats are given in Helvetica typeface.

In this document, commands and certain other "computer" lines are given in the equally-spaced Courier font. This makes characters such as blanks and periods easier to spot, and allows columns to line up as they do in computer files containing data where the column is important.

Hypoinverse is a complicated program with many features and options. Many of these "advanced" or seldom used features are documented here, but are more detailed than a typical user needs to read about when first starting with the program. I have put some of this material in smaller type so that a first time user can concentrate on the more important information.

Hypoinverse commands that take parameters are in free format, meaning that values are separated by spaces, commas or tabs. Other rules apply to free format regarding null fields, in which values are not redefined but the existing value is kept. Thus the command

```
     DUR * 3. 3* -5.2 /
```

sets the second and sixth values, but keeps the other values, including the 7th and later, as they exist.

Array sizes. The current maxima (array limits) of various data types are:

Number of stations in station file	4000
Number of stations (phase lines) per event	1000
Number of phases (P or S) per event	820
Number of homogeneous layer crustal models	36
Number of linear gradient crustal models	36
Number of crustal models of any type	36
Number of layers per crustal model	20
Number of nodes for model regionalization	124
Characters in input shadow records	103
Summary shadow records in input archive format	4

Number of fictitious station codes (UNK command):   10
Number of unknown stations to copy to output: 40

To find out the maximum sizes of the various arrays the program uses for stations, phases etc. use the MAX command. If there are too many stations ("phase cards") in an event and archive output is being generated, the excess phase data is copied to the output file without any processing and with the calculated fields left blank. These maximum array values are set by parameter statements in the source code in the common block include file, and are easy to change for situations where more or less space is needed.

Initializing with your defaults and input files

If a file called "hypinst" ("HYPINST." with no extension in VMS) is in your current directory, the file is read on startup as a command file by Hypoinverse. It may be used to set your own default values, read station or crust model files that you always use, etc. You may then enter commands directly or transfer control to other command files to do specific jobs. It is not necessary, and may be a bad idea for a new user, to use a hypinst file. I never use one because it may execute at a later date without me realizing it. Hypoinverse normally just issues a command prompt when there is no hypinst file. Any command file may be run by typing @filename.

Hypoinverse has an INI command to initialize the program by running a command file that applies to a group of people running it on a given computer. The INI command does not require that you be in the directory where the command file is located. It is for situations where a group needs to use the same parameters, station and crust model files, or where you rapidly want to initialize the program. On unix computers, Hypoinverse treats the environment variable HYPINITFILE as the command file to run when the INI command is typed. On the swave computer in Menlo Park, for example, NCSN users currently put the command

```
setenv HYPINITFILE /we/calnet/klein/hypfiles/cal2000.hyp
```

in their system startup file.

Remark codes for earthquakes and station data

Many different kinds of "remark" fields are available for input and output. We can't fully discuss all of them here, but how to use the different remark types are explained throughout this document, and the major discussions are indexed. In creative hands these are very useful.

- o Station remark (i.e. region code; 1-letter)

- o Event location remark (3-letters)

- o Event type remark, assigned by analyst (1-letter)

- o Event location problem remark, assigned by Hypoinverse (1-letter)

- o Event processing and authority remarks (three 1-letter codes)

- o Phase remark (impulsive, etc, including P first motion; 2-letters)

- o Seismogram or amplitude remark (one per phase record; 1-letter)

- o Data source codes are special remarks that are tabulated. Each phase or archive record has a 1-letter field for the data source (contributing network, timing system, etc.) Multiple data records for the same station and component can co-exist in the same event anyway, but this source code can distinguish among them. The processing for each event

determines the most common data source code for each of P times, codas and amplitudes, allowing you where the data for different events comes from.

CRUSTAL VELOCITY MODELS

All models are flat earth models with stations assumed to be at the earth's surface. Station elevations are not used, but the delaying effect of elevation can be mostly accounted for with station delays. This flat-earth assumption means that earthquake depths are relative to the average local surface defined by nearby seismic stations. This is computationally very simple.

For each crustal model, velocity can vary only with depth. Lateral variations can only be accommodated with choosing a model based on location, or by having two different models used for the same earthquake by different stations.

Multiple models

The simplest case Hypoinverse handles is one crustal model and one set of station delays used for all epicenters and all stations. Hypoinverse also allows considerable complexity by using multiple velocity models. In any model, velocity varies only with depth.

Many models may be used, each assigned to epicenters in different areas. Smooth transitions between adjacent models is accomplished by defining transition regions within which weighted averages of travel times, travel time derivatives and station delays for 2 or 3 different models are used. There is no fancy calculation of rays passing through different models, only a simple numerical weighting of travel times and travel time derivatives. Thus if the travel time for model #1 is 1.10 sec, model #2 is 1.20 sec, and the weight of model #1 is 20% (and model #2 is 80%), the travel time used is 1.12 sec. It is that simple. A weighted average of emergence angles is also calculated, but is not used in the hypocenter solution.

The geometry of assigning models to different areas consists of a list "nodes" or points on a map. Each node is assigned to a model along with the radius of a circle within which that model is used. Several nodes may be assigned to the same model, and if so the circles can and often do overlap. It is thus possible to define an irregularly shaped region as the union of several circles. Each node is defined by a NOD command.

This is algorithm for determining the crustal model(s) used: If an epicenter lies within the inner circle surrounding any node, that model is used exclusively. An outer circle must also be defined for each node, which describes how far out its influence extends. If an epicenter lies between inner and outer circles, it receives a partial "weight" for that model. The weight is a smooth cosine taper between inner circle (weight 1) and outer circle (weight 0). For a given earthquake location, each node is tested (in the order it was defined) to see if the epicenter lies within its outer circle, but testing stops when the epicenter is found inside 3 outer circles. If the weights total more than 1.0, they are normalized to 1.0. If they total less than 1.0, the difference is made up using the default model (model number 1) such that the weights always total 1.0. If the epicenter lies outside all circles, the default model is used exclusively. The mix of models is determined for each iteration and the epicenter may migrate from one model to another.

Figure 1 is a hypothetical example of two crustal models (in addition to the default model used for all surrounding regions, shown in white) defined by 4 nodes. These are the commands that might be used to define this model geometry:

```
MUL  T 1            Enable multiple models, assign model 1 as the default model.

NOD  37 38 122 24 10 10 2

NOD  37 35 122 24 10 10 2

NOD  37 32 122 24 10 10 2
```

These three NOD commands establish the centers (in deg and min of latitude and longitude), a 10 km radius of the inner circle (dark gray), a 10 km wide transition zone surrounding each inner circle (light gray), and assign the node to model number 2. The region assigned exclusively to model 2 is the union of the three inner circles, hence the sausage shape to the dark gray area. One node defines the circular area for model number 3, with the same 10 km radius:

```
NOD  37 35 122 18 10 10 3
```

The transition regions surrounding the inner circles are where things get interesting, as up to three models are used, each with a weight less than 1.0. Hypoinverse uses the algorithm described above to find the weights for up to 3 nodes for earthquakes in the transition (outer circle) regions such that all weights total 1.0.

Model 1 100% (default model)
Weighted mixture of models 1 & 2
Model 2 100%
Weighted mixture of models 1, 2 & 3
Weighted mixture of models 1 & 3
Model 3 100%

"Node" point, the center of an inner and outer circle.
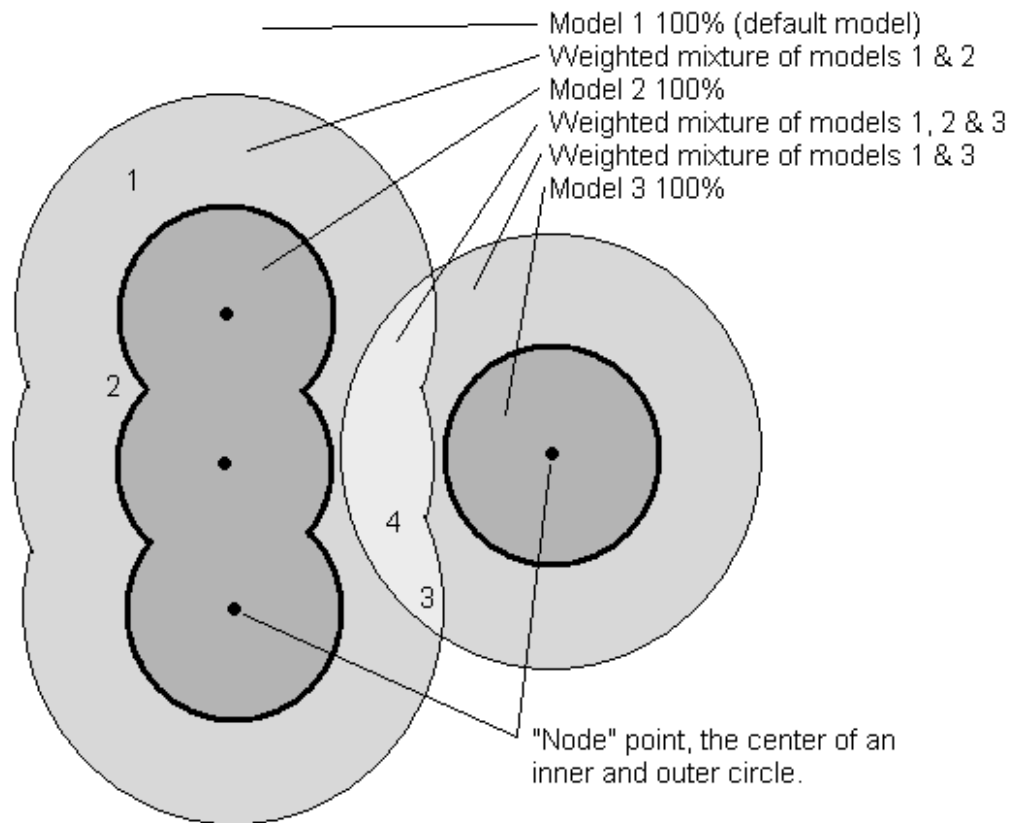
Figure 1. A example of nodes defining the areas for two crustal models and the transition regions between them.

Earthquakes in the dark gray areas within the inner circles use one model exclusively. Earthquakes in the white area surrounding all of the circles use the default model exclusively. Earthquakes in the lighter gray transition regions use a weighted average of models. Numbers indicate locations where the weighting of different models is discussed in the text.

Consider these example cases for the numbered locations in figure 1: 1) A simple location that is only in one of the node's outer circle regions. Following the algorithm, point 1 gets about _ weight for model #2. Because the total weight is less than 1.0, the remaining weight to total 1.0 is made from the default model #1. Thus only models 1 and 2 are used in this area, with a cosine taper to make the transition smooth. 2) This point is just outside the inner circles for two nodes, both assigned to model #2. The weight for each node might be 90% to total 1.80. Because this number is greater than 1.0, it is normalized to 1.0 and model #2 gets full weight, 50% from each node. 3) This is near the outer boundary of two outer circles assigned to models #2 and #3. It might get 15% weight from each of models 2 and 3 to total 0.30. The remaining weight to reach 1.0 would be 70% and would be from the default model #1. 4) This point is within the outer regions of 3 nodes and might get 30-40% weight from each. If the cosine taper value is 30%, the total weight from all 3 nodes is 0.90. Then model #1 contributes 10%, model #2 60% and model #3 30%. If the cosine taper value from each node is 40%, the total is more than 1.0, and the normalized contributions are 67% from model #2 ad 33% from model #3. This example does not have a point where the outer regions from 4 different nodes all meet, but if it did, only the first three encountered in the list of nodes would be used to determine weights.

Some simple guidelines in defining the nodes and their surrounding circles will help avoid location problems and sharp transitions between models that can form artificial discontinuities. 1) Never have the inner circles for different models overlap. This will make a sharp discontinuity between models. For a given epicenter location, each node is tested in the order it was defined to see if the epicenter lies in its inner circle. If it does lie in an inner circle, it is assigned 100% to that model even if it is also in the inner circle for a later node. 2) It is best to have the inner regions surrounding nodes be the high seismicity areas such as faults, and the transition regions between models be areas of low seismicity. One then eliminates the complex mixing of different models from influencing the patterns of seismicity. 3) Allow wide transition zones between the inner circles of different models. In other words, avoid having 10 km radius inner circles with 1 km wide transition zones. This will cause rapid or even sharp transitions between models that may cause epicenters to locate on arcs or form other strange patterns.

Figure 2 shows a sample of regions defined for Northern California. The transition zones between models are probably too narrow in many places, but they are generally locations of low seismicity.

Each crustal model is read separately with a CRH or CRT command. This associates each model with a model number and a 3-letter code (the beginning of the model name). This 3-letter code labels the model in all of the output files. Each model has its own set of station delays. The DEL command reads a file containing station codes and delays, one station per line. The DEL command also associates the delays in the file with a model number. You may have a different delay file for each model (recommended) or put the delays for all models in one file. If all delays are in one file, the column a delay uses must correspond to the model number used for that model when the crustal files are read. I find it easier to keep one model's delays in one file. Models are labeled on output by 3-letter codes and not their number. The MUL command is used to select either single model or multiple model modes and to define the default model.
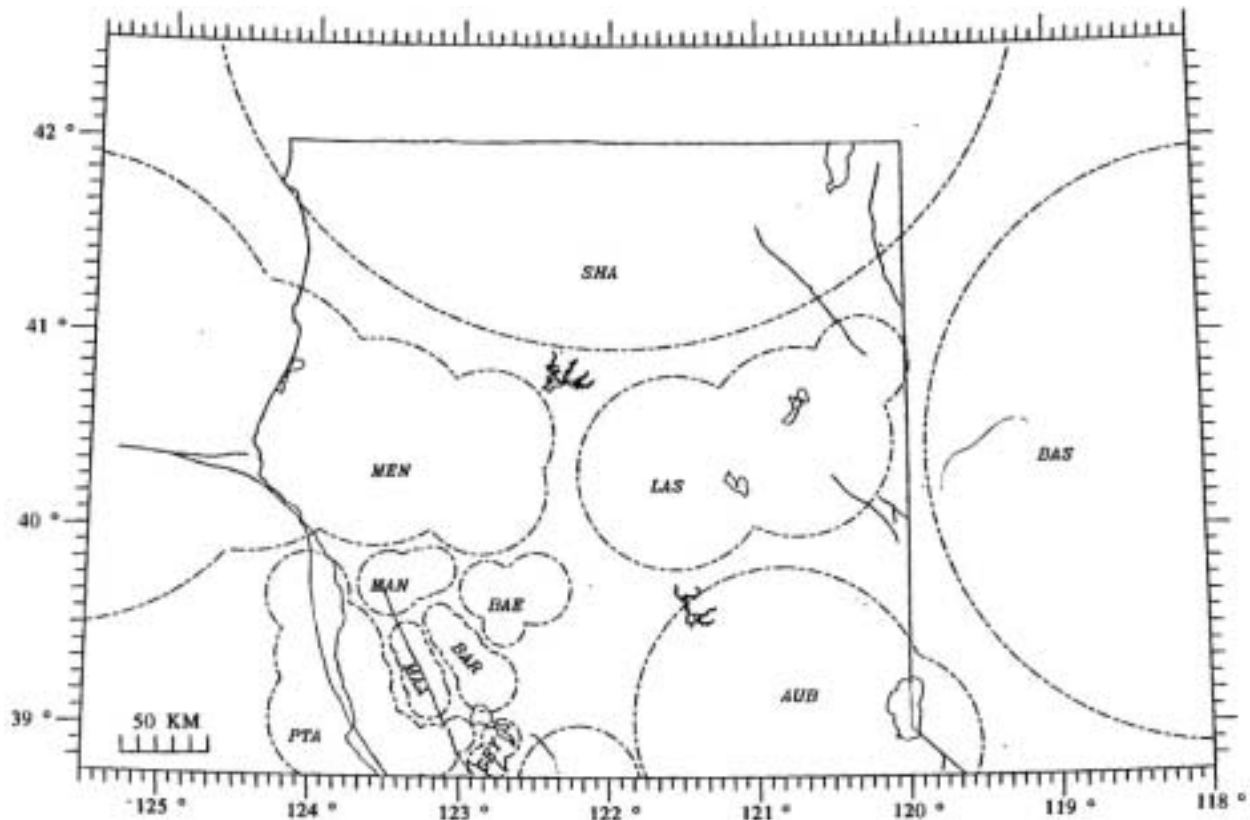
Figure 2. An example of regions for mutiple crustal models.

Model regions are defined by the union of circles. Each group of circles is a different model. The circles are defined by NOD commands. Only the inner circles are plotted (see text), and the regions between model regions use a weighted combination of models, perhaps including the default model.

Alternate models

Hypoinverse also has an alternate model capability permitting the use of different models by different stations for the same hypocenter. A typical use of alternate models is where the crust is different on each side of a fault (like the San Andreas), earthquakes occur on or near the fault, and the rays to all stations on each side of the fault should use their own model. Typically this alternate model feature is most useful only where two models and sets of delays have been derived for two station sets. *Alternate* models are most useful where a single vertical discontinuity in the crust is known. *Multiple* models are useful where different regions have locally determined models and the gradations between models are either smooth or unknown. Keep in mind that station delays can also be adjusted to shift epicenters laterally or move epicenters on or off a fault. The combination of alternate and multiple models, each with their own sets of delays, can become quite complex, is very flexible, but can possibly be misused.

To use the alternate model feature, read in two models using any two different model numbers. Then use the ALT command to designate the numbers of the primary and alternate models, which then form a pair. The station file or an "alternate-station delay file" must designate which stations use the alternate model. Alternate models may be used with or without the multiple-model feature. If you use both together, be sure to designate the primary and not the alternate

model number in the NOD commands because the same set of nodes will be used for both models in the pair. Several different models may have alternates, but the same set of stations must use alternate models in all cases. Thus, a given station will be either "primary" or "alternate" for all of the "primary" and "alternate" models. An alternate model must be of the same type (layer or gradient) as the primary model.

Model types: homogeneous layer and linear gradient

Models may be of two different types, which are stored and calculated differently. The simplest is the homogeneous layer model, which calculates travel times directly from the velocity structure. The second model type uses layers with linear velocity gradients, but requires that a travel-time table be generated previously by the program TTGEN. The table needs to be generated only once, and Hypoinverse uses it very efficiently by merely interpolating from it to get all travel times and derivatives. Tests on the VAX computer show that using a travel-time table requires about 60% of the CPU time used with layer models. Use the CRH command to read homogeneous layer models and the CRT command for reading gradient (travel-time table) models. The two model types may be used simultaneously.

Homogeneous layer models

Each model may consist of up to 20 homogeneous layers including the half-space. Velocity must increase with depth. Use the CRH command to specify the model number and the name of the file containing the homogeneous layer model. The CRH command also reads the model into memory. For example:

```
CRH 2 'CRUST2.CRH'
```

The format of the homogeneous layer crust model file

('CRUST2.CRH' in the example) is:

Line	format	data
Line 1:	A30	Model name.
Lines 2 and later:	2F5.2	Velocity of layer and depth to its top.

The first 3 letters of the 30-letter model name are used as the model code that appears in the print, summary and archive outputs. Use one line per layer, top layer first. The depth to the top of first layer must be 0.0, and the last layer is the underlying half-space.

Linear gradient models using a travel time table

An alternative and more computationally efficient way to compute travel times is by interpolation within a table generated prior to running Hypoinverse. Use of a table permits more complex travel time calculations, such as linear velocity gradients within layers and capacity for a buried low-velocity zone. The travel-time table must be calculated and written to a file using the program TTGEN (appendix 5) prior to locating earthquakes.

A travel-time table may be calculated for a velocity-depth function consisting of from 2 to 20 points at which the velocity and depth are specified. The velocity is then assumed to be linear between points, i.e., with a uniform gradient within layers. Several restrictions apply to the possible models: (1) No two velocity-depth points may be at the same depth (a sharp velocity discontinuity is not allowed). Discontinuities may be modeled using thin layers with high gradients, but the transition layer should be thick enough that one or two rays used to generate the travel-time table will bottom within the layer and define a reverse branch of the travel time curve. (2) The depth of the first point must be 0.0, and other points must be given in increasing order of depth. (3) The last (deepest) point sets the velocity of the homogeneous half space assumed to underlie the model. (4) The half-space velocity must be the greatest of any specified to insure that rays can be refracted along the top of the half-space. (5) One buried low-velocity zone is permitted in each model, i.e. velocity may not decrease with depth except for one group of adjacent layers. (6) Assigning the same velocity to two adjacent points may specify homogeneous layers.

Use the CRT command to specify the model number and the file name containing the travel-time table to be assigned to that model. The CRT command also reads the model into memory. The first 3 letters of the model name (as originally assigned before running TTGEN) are used as the model code for labeling output. For example:

```
CRT 1 'MODEL1.CRT'
```

Reading crust model files is more efficient if they are read in binary instead of ASCII. You may create a binary crust model file after reading in all crust model data including station delays and the multiple model parameters. Use the WCR command to write a snapshot of the crust model arrays to a binary file. Read the file back in with the RCR command. The RCR command replaces the CRH, CRT, MUL, ALT and NOD commands. Tests show that binary reads using the RCR command are at least five times faster than equivalent ASCII reads, and the speedup increases with more models.

THE STATION LIST AND USE OF STATION DELAYS, GAIN HISTORY AND MAGNITUDE CORRECTIONS

Station location file

Specify the file containing names, coordinates and other station data using the STA command. For example:

```
STA '1984.STA'
```

The station data is read into memory as soon as this command is given, and is kept until another STA command is issued. The station file must contain one line per station. Use the H71 command to select either Hypoinverse (old style or new style) or HYPO71 file format. There is no separateY2000 format because the files do not contain dates.

Reading station files is more efficient if they are read in binary instead of ASCII. You may create a binary station file after reading in all station data including the multiple model delays. Use the WST command to write a snapshot of the station arrays to a binary file. Read the file back in with the RST command. The RST command replaces the STA and DEL commands. The RST command does not replace the XMC or FMC commands to read magnitude corrections. If used, the XMC and FMC commands must be given after RST because the magnitude corrections are applied to stations already in memory. The RST command does not replace the ATE or CAL commands to manage gain histories. If a calibration factor is read from the station line with the STA command, it is written to the binary

file. The calibration history and expiration dates, however, are not written to the binary file. If you are using attenuation histories, issue the ATE command after RST to read and dynamically update the attenuation's. Similarly, use the CAL command for cal factor histories. Tests show that binary reads using the RST command are several times faster than equivalent ASCII reads.

Station name codes and file formats

There are 3 different station file formats currently supported. They differ in the amount of station information supplied and in the number of letters allowed in the station name code. The HYPO71 format is supported for backwards compatibility. HYPO71 allowed only 4 letters. Hypoinverse format 1 maintained the 4-letters, but added other letters on in various places as needed. Hypoinverse format 2 supports the full 12-letters and thus supports the full IRIS and SEED specifications, and stores locations to 0.0001 minute. The H71 command chooses the format to be used.

The complete code consists of a 5-letter site code assigned by each network, a 2-letter net code assigned each network by Tim Ahern at IRIS, a 3-letter component or channel code for different records at each site, and a 2-letter location code to further discriminate channels. 12-letters (with the 2-letter "location code" extension) are now available because of the proliferation of stations and the need to avoid duplication. Hypoinverse also stores a 1-letter component field as an abbreviation to the full 3-letter field because this has been a common historical practice, and because 3-letter components are often confusing to people. Output files carry both the 1-letter and 3-letter components. Although conceptually different, we have often mixed letters representing these 3 types in the same field for practicality. Thus CAL, CALE and CALN may represent 3 components at the same site. Hypoinverse allows use of the full 12 letters or a subset of each of the 4 fields. The LET command chooses how many letters from each field you want to use.

The newest station name field is the 2-letter location code. This code discriminates between different channels that would otherwise have identical site, net and component codes. Examples include the same instrument recorded by different data loggers, similar instruments at the site, a station that was moved to different location but not given a new code, or an experiment type code. It can function like an extension of the channel or site code.

The LET command specifies how many letters of the 2-letter location code you want to test when matching stations in the station file with stations in other files. In the LET command you separately state how many location letters to use (0-2) for matching station names in 1) phase files (NSLOC) or 2) other station files (like magnitude correction or cal factor; NSLOC2). For example, say many stations were moved a few hundred feet to a better site, but not given a new code at the time. If you do not want to implement location codes, use 0 as the number of location letters to compare, and the location codes will be ignored. Alternatively, each site could get a different location code (like '01' and '02'), and comparing 2 letters with NSLOC=2 would use location codes and the appropriate station location to compute the earthquake solution. This is because 2 letters are used to match location codes in station and phase files, and station ABC.01 will not match station ABC.02. Because the station was moved slightly (and thus both location codes did not operate at the same time) you can use the same gain history and magnitude correction files for both locations by setting NSLOC2=0. Thus station ABC in the *magnitude correction file* matches both ABC.01 and ABC.02 in the station file, and the correction is used for both "stations". If you put magnitude corrections for each station in the *station file*, matching is not an issue and it does not matter what NSLOC2 is.

Note that if you read station delays from delay files (and not in the station file), a delay will always match and be assigned to all components and all location codes. In other words, the delay for ABC will be assigned to ABC.V.01, ABC.V.02 and ABC.E.

If a non-blank location code is used on input in the station file, it will be passed to the output archive file. This is true even if the location code is not used to match stations in the station and phase files. Note this warning: If two stations have identical codes except for different location codes, their data can be treated independently and output to the archive file properly <u>only if</u> location codes are compared with NSLOC > 0. If you do not check location codes (NSLOC=0 or NSLOC2=0), station picks or other data will be applied to the <u>first</u> of the matching station entries regardless of location code, of the two or more otherwise identical entries. This means that the first location code in the station file will be used on all output archive and print files regardless of which location code was input.

This can generate errors. It is thus best to fill only the station code fields that are actually being compared with the lengths given in the LET command.

Station delays will be applied correctly to all stations with the correct site and net codes regardless of component or location codes.

The only restriction on naming stations is that the first four characters of the 5-letter site code must not be all blank. This is to recognize the terminator record at the end of the event. It is suggested that the component code be used for the component, rather than combining the component with the site code as done previously. This to take advantage of certain special uses of the component code, such as applying a delay to all components at a site, and using station magnitudes only from certain components to average in an event magnitude.

Hypoinverse requires a separate station input line for each component for each station. This makes handling the database much simpler, even though components at the same site generally share common information, such as location or delay. The station file is read in, and each line is treated independently. When a phase line is read, Hypoinverse searches the station list from the top until a match of the code is found. This matching test uses the number of letters in each station code that are specified with the LET command. Any later station with the same code is ignored. You can choose how many letters (including 0) of each of the 4 fields to compare when doing this search. If your net field is blank on both station and phase lines, for example, it does not matter how many letters (0, 1 or 2) you compare because you will always get a match. See the LET command. Note, as with the location code, that if a net code is supplied in the station file but not compared (because the number of comparison characters is set to 0 by the LET command), the first matching NET code will be output to the archive file regardless of input NET code.

The exception to the requirement that all 4 station fields match is as follows: after you read in the station location file, you can read in a separate delay file. When matching a delay to a station in memory, only the site and net codes are compared. Searching continues through the station list to assign the delay to all components and location codes at that site.

These are the lengths of station name fields in different formats (later extensions are shown as +)

	Site code	Net code	Component	Location	Separate 1-letter comp.
HYPO71	4	-	-	-	no
HI #1	4	0 + 2	0 + 3	0+2	yes
HI #2	5	2	3	2	yes

Specifying instrument types and calibration factors

These comments apply to all three station formats and are given here in one place. See the discussion of cal factors in the amplitude magnitude section and reading in the calibration and attenuation history files section for more information. The calibration factor is used for amplitude magnitudes and is a measure of station gain relative to a standard seismograph. The calibration factor was originally defined as the peak-to-peak amplitude of a 10 microvolt RMS signal at 5 hz applied to the VCO and measured in mm on the Develocorder film viewer. For instrument types 0 and 2 the cal factor should generally be 1.0. A cal factor of 0.0 signifies an unknown response for which no amplitude magnitudes will be computed. If a cal factor is given on a phase line (traditional format only) it overrides the value supplied for the station.

The instrument type specifies which frequency response curve and which seismometer motor constant applies to the instrument. The response correction yields an equivalent Wood Anderson amplitude. The seismometer types are listed in the instrument types table in the amplitude magnitude section below.

The VCO (preamplifier in the voltage controlled oscillator) attenuation may be given in place of the cal factor (see the ATN command). An entire history of station attenuations with the dates of attenuation changes may be read from a separate file with the ATE command. Equivalently, an entire history of station cal factors with the dates of changes may be read from a separate file with the CAL command. These history files are read as necessary during a location run for updates to the calibration history with time. To know which (cal factor or attenuation) of the history files to read for the analog stations, the station is given an instrument type of 1 for attenuation history and 3 for cal factor history.

The HYPO71 station data format

Start Col.	Len.	Fortran Format	Data
1	1	A1	Optional 1-letter station component code.
2	1	A1	Station weight code (in units of 0.1) by which the weights assigned each P & S phase are to be multiplied. Use the digits 0-9 for the weight in tenths; "*" or "0" for no weight; or any other character (including blank) for full weight.
3	4	A4	Station name code. The code may not be blank. The first character may not be the $ character. See also col. 1.
7	2	I2	Latitude, degrees.
9	5	F5.2	Latitude, minutes.
14	1	A1	N or blank for north latitude, S for south.
15	3	I3	Longitude, degrees.
18	5	F5.2	Longitude, minutes.
23	1	A1	W or blank for west longitude, E for east.
24	4	4X, 1X	Reserved for elevation in m. Not used by Hypoinverse.
29	5	F5.2, 1X	P delay (sec) for delay set 1.
35	1	A1, 2X	Optional station remark field to copy to print output.
38	5	F5.2, 2X	Duration magnitude correction.
45	5	F5.2, 1X	Amplitude magnitude correction.
51	1	I1, 1X	Instrument type code.
53	4	F4.2, 1X	Default period (in sec) at which the maximum amplitude will be read for this station. Must be greater than 0.1
58-63	6	F6.2	Default calibration factor for amplitude magnitudes.

The HYPOINVERSE station data format #1

Start Col.	Len.	Fortran Format	Data
1	4	A4	Station site code. The first character may not be the $ character.
5	1	A1	Station weight code (in units of 0.1) by which the weights assigned each P & S phase are to be multiplied. Use the digits 0-9 for the weight in tenths; "*" or "0" for no weight; or any other character (including blank) for full weight.
6	2	I2, 1X	Latitude, degrees.

Col.	Len.	Format	Data
9	5	F5.2	Latitude, minutes.
14	1	A1	N or blank for north latitude, S for south.
15	3	I3, 1X	Longitude, degrees.
19	5	F5.2	Longitude, minutes.
24	1	A1	W or blank for west longitude, E for east.
25	4	4X	Reserved for elevation in m. Not used by HYPOINVERSE.
29	3	F3.1	Default period (in sec) at which the maximum amplitude will be read for this station. Must be greater than 0.1.
32	1	A1, 1X	Optional 1-letter station component code.
34	1	A1	Put a "2" or "A" here to designate this as an alternate crust model station. Both alternate and primary crustal models must be in use. Stations may also be tagged for use with an alternate model in the delay file.
35	1	A1	Optional station remark field to copy to print output.
36	5	F5.2, 1X	P delay (sec) for delay set 1.
42	5	F5.2, 1X	P delay (sec) for delay set 2.
48	5	F5.2	Amplitude magnitude correction. If in the range +/-2.4, the correction is included (by addition) in the amplitude magnitude, If you don't want a station's magnitude used in the event magnitude, use a correction of 5.0 plus the actual correction. You can also assign a zero weight (see next).
53	1	A1	Amplitude magnitude weight code. Codes 0-9, "*" and blank are used the same as the P & S weight codes (col 5). The actual magnitude weight used is the product of those on the station and phase lines. See also col 48.
54	5	F5.2	Duration magnitude correction (works the same as the amplitude magnitude correction).
59	1	A1	Duration magnitude weight code (works the same as the amplitude weight code).
60	1	I1	Instrument type code.
61	6	F6.2	Calibration factor.
67	2	A2,1X	Seismic network code.
70	3	A3	3-letter component or channel code.
73-74	2	A2	2-letter location code (site/component extension).

The HYPOINVERSE station data format #2

Start Col.	Len.	Fortran Format	Data
1	5	A5,1X	Station site code. The first character may not be the $ character.
7	2	A2,1X	Seismic network code.
10	1	A1	Optional 1-letter station component code.
11	3	A3,1X	3-letter component or channel code.
15	1	A1	Station weight code (in units of 0.1) by which the weights assigned each P & S phase are to be multiplied. Use the digits

			0-9 for the weight in tenths; "*" or "0" for no weight; or any other character (including blank) for full weight.
16	2	I2, 1X	Latitude, degrees.
19	7	F7.4	Latitude, minutes.
26	1	A1	N or blank for north latitude, S for south.
27	3	I3, 1X	Longitude, degrees.
31	7	F7.4	Longitude, minutes.
38	1	A1	W or blank for west longitude, E for east.
39	4	4X	Reserved for elevation in m. Not used by HYPOINVERSE.
43	3	F3.1,2X	Default period (in sec) at which the maximum amplitude will be read for this station. Must be greater than 0.1.
48	1	A1	Put a "2" or "A" here to designate this as an alternate crust model station. Both alternate and primary crustal models must be in use. Stations may also be tagged for use with an alternate model in the delay file.
49	1	A1	Optional station remark field to copy to print output.
50	5	F5.2, 1X	P delay (sec) for delay set 1.
56	5	F5.2, 1X	P delay (sec) for delay set 2.
62	5	F5.2	Amplitude magnitude correction. If in the range +/-2.4, the correction is included (by addition) in the amplitude magnitude, If you don't want a station's magnitude used in the event magnitude, use a correction of 5.0 plus the actual correction. You can also assign a zero weight (see next).
67	1	A1	Amplitude magnitude weight code. Codes 0-9, "*" and blank are used the same as the P & S weight codes (col 15). The actual magnitude weight used is the product of those on the station and phase lines. See also col 62.
68	5	F5.2	Duration magnitude correction (works the same as the amplitude magnitude correction).
73	1	A1	Duration magnitude weight code (works the same as the amplitude weight code).
74	1	I1	Instrument type code.
75	6	F6.2	Calibration factor.
81-82	2	A2	2-letter location code (site/component extension)

Station delay file

The Hypoinverse station file format holds two delays for models 1 and 2 (the HYPO71 format holds one delay). If you need more than the 1 or 2 delays for use with multiple crustal models (see preceding section), you must read station delays from a separate delay file using the DEL command. Each line of the delay file has the station code and one or more delays.

The alternate model code for stations may either be in the station file or a file only containing codes for the alternate stations. A station becomes an alternate if so designated in either file.

Only the station site and network codes are read in the delay files. All components at a site are assumed to have the same delay, and the delay will be assigned to all components at that site. One could get around this assumption (for a borehole station for example) by using a different site code. When matching the station codes in the station

location file and the station delay file, you can use less than the full 5-letter site code and full 2-letter net code. See the LET command.

There are three types of delay file read with the DEL command. The file type is determined by the first argument of the DEL command: -1 reads a file with alternate codes (designating which stations use the alternate model set with the ALT commands), 0 reads a file with delays for all models, and the positive model number (1-40) reads delays for the model number specified by the DEL command.

The formats of the delay files are:

Delay file for one model

(`DEL n 'filename'`, n is the model number)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2, 1X	Station net code. Delays are for all components at this site.
10	5	F5.2, 1X	P delay for model n.
16	1	A1	Optional source code for delay (not read by Hypoinverse). Codes presently in use are: blank – delay derived for this model, N - delay from a nearby station, A - delay assumed from a slightly different crustal model, B - delay borrowed from a nearby region.

Delay file for all models

(`DEL 0 'filename'`)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2, 2X	Station net code. Delays are for all components at this site.
11	1	A1	Put an "A" here to use alternate crust models with this station.
12		40F4.2	P delays for each model in order of crust model number.

Alternate model file

(`DEL -1 'filename'`)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2, 2X	Station net code. Alternate status is for all components at this site.
11	1	A1	Put an "A" here to use alternate crust models with this station.

You <u>must</u> read in the station list using the `STA` command before reading delays with the `DEL` command. The station file contains all the stations you will use for locations and the `STA` command reads these into memory. The delay file may contain any set of stations: delays for stations already in memory will be used and those for a station not in the station file will be ignored. The delay file may thus be a master list containing all known delays and the stations may be in any order. Delays for a station in the station file but not the delay file will have delays left at 0, or as read in the station file for models 1 and 2.

Station attenuation and calibration factor files

The station gain curve is specified by a standard response curve for that instrument type and by a calibration factor that fixes the level of the curve at a particular frequency. See the section on amplitude magnitudes for a fuller discussion. In the following discussion "attenuation" applies primarily to analog stations with a preamplifier and VCO located at the seismometer. These are typical NCSN stations but are widely copied by other networks. The pre-amplifier located at the seismometer has a gain of 90db, from which an attenuator lowers the net gain in increments of 6 db. You can thus use either the calibration factor or the attenuator setting to specify gain.

There are five options for specifying station gain information for use in calculating amplitude magnitudes or correcting coda magnitudes: 1) put the calibration factor in the station file to use for the whole location run; 2) specify the pre-amplifier attenuation in place of the station calibration factor on the station line (see the ATN command); 3) put calibration factors on the phase line (traditional phase format only) to use only for that event and override that on the station line if present (this will not preserve the calibration or write it out to the archive file. This feature is for backward compatibility and is not recommended); 4) read the attenuation history from a separate file and extract the attenuation for any given date (see the ATE command); and 5) read the calibration history from a separate file and extract the calibration factor for any given date (see the CAL command).

Using a separate station attenuation or calibration file insures that the correct gain information will be used for each station on the date of the earthquake. Each attenuation or calibration factor value has an associated expiration date. When an event is being processed whose date is after the attenuation's expiration date, Hypoinverse rereads the file and uses the updated value. The cal or attenuation file has one station per line. Unlike the delay file, full 12-letter station codes are supported because the attenuation depends on the component. The line consists of a station code followed by pairs of attenuation or cal values and their expiration dates. The last attenuation on a line must have its expiration date left blank or set to zero to indicate that it applies into the future. A station history may have at most 7 attenuations. If the attenuation for a station never changed, the line consists only of the station code and one attenuation value. The attenuation is in db and is a multiple of 6 (0, 6, 12...60). The old format does not have the space for the century but the Y2000 format does. The choice of which format is used depends on the selection made with the "200″ command.

Specifying a calibration factor of 0 (or leaving the field blank) means that no magnitudes will be computed for this station, even if an amplitude is supplied.

The relation between the CAL factor and attenuation setting is

$$\log(\text{CAL}) \sim= -0.05*\text{atten} + 1.35$$

atten:	0	6	12	18	24	30	36	42	48	56	60
cal:	25.2	11.65	5.576	2.795	1.4	0.702	0.352	0.176	0.0885	0.044	0.0222

atten:	66	72	78
cal:	0.0111	0.00557	0.00279

Attenuation history file format (pre Y2000)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2,2X	Station net code.
11	3	A3,1X	Station component code.
15	2	I2, 1X	First attenuation setting. Must be a multiple of 6 db.
18	8	4I2, 3X	Y, M, D, H expiration date and hour of first attenuation. Minutes, if supplied, will not be used. Leave the expiration of the last attenuation blank to indicate no expiration.
29	2	I2, 1X	Second attenuation.
32	8	4I2, 3X	Second expiration date.

Format repeats: 7(I2, 1X, 4I2, 3X)

Attenuation history file format (Y2000 century compatible)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2,	Station net code.
9	2	A2,	Station location code.
11	3	A3,1X	Station component code.
15	2	I2, 1X	First attenuation setting. Must be a multiple of 6 db.
18	4	I4	Year of expiration of first attenuation.
22	6	3I2, 1X	M, D, H expiration date and hour of first attenuation. Leave the expiration of the last attenuation blank to indicate no expiration.
29	2	I2, 1X	Second attenuation.
32	4	I4	Second expiration year.
36	6	3I2, 1X	Second expiration date M, D, H.

Format repeats: 7(I2, 1X, I4, 3I2, 1X)

Use the ATE command to read the station attenuation file. The phase data file must be in chronological order to insure getting correct attenuations if there is ever more than one attenuation value per station. You must read your station file (using the STA command) before the attenuation file (with the ATE command) because Hypoinverse stores attenuations only for the stations already read into memory. This means that the attenuation file can contain the history of the entire network and only the data needed will consume space in Hypoinverse. Note

that the instrument type must be 1 for stations with attenuations supplied with the ATE command.

The filename given with the ATE command is read both when the ATE command is given and as necessary to update an attenuation. The ATE command also asks for a date and time for which to load the initial attenuations. If you use the date of the first earthquake you wish to locate, Hypoinverse won't have to waste much time re-reading the attenuation file. If you specify a year of 0, Hypoinverse loads the earliest attenuation for each station. This will require more updates from the attenuation file but will not require knowing the date of the first event. This may account for taking several extra seconds to locate your first earthquake. Attenuations are converted directly into calibration factors when read in.

Calibration factor history file format (pre Y2000)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2, 2X	Station net code.
11	3	A3, 1X	Station component code.
15	7	F7.2, 1X	First calibration factor.
23	8	4I2, 3X	Y, M, D, H expiration date and hour of first calibration factor. Minutes, if supplied, will not be used. Leave the expiration of the last calibration factor blank to indicate no expiration.
34	7	F7.2, 1X	Second calibration factor.
42	8	4I2, 3X	Second expiration date.

Format repeats: 7(F7.2, 1X, 4I2, 3X)

Calibration factor history file format (Y2000 century compatible)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2,	Station net code.
9	2	A2,	Station location code.
11	3	A3, 1X	Station component code.
15	7	F7.2, 1X	First calibration factor.
23	4	I4	Year of expiration of first calibration factor.
27	6	3I2, 1X	M, D, H expiration date and hour of first cal factor. Leave the expiration of the last cal factor blank to indicate no expiration.
34	7	F7.2, 1X	Second calibration factor.
42	4	I4	Second expiration year.
46	6	3I2, 1X	Second expiration date M, D, H.

Format repeats: 7(F7.2, 1X, I4, 3I2, 1X)

Use the CAL command to read the station calibration factor file. The phase data file must be in chronological order to insure getting correct calibration factors if there is ever more than one value per station. You must read your station file (using the STA command) before the calibration factor file (with the CAL command) because Hypoinverse stores calibration factors only for the stations already read into memory. This means that the cal factor file can contain the

history of the entire network and only the data needed will consume space in Hypoinverse. For analog stations with cal factors supplied with the CAL command, note that the instrument type must be 3. The CAL command (and <u>not</u> the ATE command) is appropriate for digital stations.

Station magnitude correction files

Hypoinverse calculates independent coda and amplitude magnitudes and uses separate corrections for each. There are two options for specifying station magnitude corrections: 1) put the magnitude correction in the station file to use for the whole location run; or 2) read the magnitude corrections from a separate file and assign the corrections to stations already in memory. Duration and amplitude magnitude corrections may both vary with time. This might be because some changes are not reflected in the attenuation or calibration history. The ability to change magnitude corrections with time can also be used to turn a station's magnitude weight on and off with time (change weight between 0 and 1). The XMC command reads the amplitude magnitude corrections and FMC reads the duration magnitude corrections for a specified date. The XMC command can also assign instrument types to stations.

You must read your station file (using the STA command) before the magnitude correction files because Hypoinverse stores corrections only for the stations already read into memory. This means that the magnitude correction file can contain the history of the whole network and only the data needed will consume space in Hypoinverse.

The handling of magnitude corrections is very similar to that for station attenuations and cal factors. Using a separate magnitude correction file insures that the correct information will be used for each station on the date of the earthquake. Each FMAG (duration) or XMAG (amplitude) correction has an associated expiration date. When an event is being processed whose date is after the correction's expiration date, Hypoinverse rereads the file and uses the updated correction. Both FMAG and XMAG correction files have one station per line. Unlike the delay file, full 12-letter station codes are supported because the magnitude correction depends on the component and location codes.

The file consists of a station code followed by pairs of magnitude corrections and their expiration dates. The last correction must have its expiration date left blank or set to zero to indicate that it applies into the future. A station may have at most 6 corrections each for XMAG and FMAG. If the magnitude correction for a station never changed, only one correction is needed and no expiration date is needed. Use the FMC command to read the FMAG correction file, and the XMC command to read the XMAG correction file. If there is more than one correction per station, the phase data file must be in chronological order to insure getting correct magnitude corrections.

Normally the magnitude weight that governs whether the station magnitude is used in the event magnitude, is taken from the station file. Setting an optional flag in the FMC or XMC commands (to false) gives all stations <u>not</u> in the correction file a zero weight. Setting that flag also gives <u>all</u> stations in the correction file full weight, regardless of the weight in the station file. Thus it is possible to ignore magnitudes from stations that don't have well-determined corrections. Most networks set the flag in the FMC or XMC commands to true, and leave the station weights in the station blank to give full weight.

You can use magnitude corrections to give zero weight to magnitudes from individual stations when computing the event magnitude. Just add 5.0 to the actual correction to suppress that station's magnitude contribution in the event magnitude. This works for both amplitude (XMC)

and coda (FMC) magnitudes. If you want to suppress the gain correction (coda magnitudes only) for individual stations, add 10.0 to the coda magnitude correction. These two flags for coda magnitudes function independently: if you want to zero weight <u>and</u> not make the gain correction to a coda magnitude, add 15.0 to the correction.

The filename given with the FMC or XMC command is read both when the command is given and as necessary to update a correction. The FMC and XMC commands also ask for a date and hour for which to load the initial magnitude corrections. If you use the date of the first earthquake you wish to locate, Hypoinverse won't have to waste much time rereading the file. If you specify a year of 0, Hypoinverse loads the earliest magnitude correction for each station. This will require more updates from the correction file but will not require knowing the date of the first event.

Use the FMC command to read the FMAG correction file. The choice of which Y2000 format is used is governed by the "200" command. Corrections can also suppress the weight of individual stations by adding 5.0 (see text above). FMAG corrections can also suppress the gain corrections by adding 10.0.

Duration magnitude correction file format (pre Y2000)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2, 2X	Station net code.
11	3	A3, 1X	Station component code.
15	5	F5.2, 1X	First magnitude correction.
21	2	I2	Year of expiration of first magnitude correction.
23	6	3I2, 3X	M, D, H expiration date and hour of first magnitude correction. Leave the expiration of the last magnitude correction blank to indicate no expiration. Minutes, if supplied, is ignored.
32	5	F5.2, 1X	Second magnitude correction.
38	2	I2	Second expiration year.
40	6	3I2, 3X	Second expiration date M, D, H.

Format repeats: 6(F5.2, 1X, I2, 3I2, 3X)

Duration magnitude correction file format (Y2000 century compatible)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2	Station net code.
9	2	A2	Station location code.
11	3	A3, 1X	Station component code.
15	5	F5.2, 1X	First magnitude correction.
21	4	I4	Year of expiration of first magnitude correction.
25	6	3I2, 1X	M, D, H expiration date and hour of first magnitude correction. Leave the expiration of the last magnitude correction blank to indicate no expiration.
32	5	F5.2, 1X	Second magnitude correction.

| 38 | 4 | I4 | Second expiration year. |
| 42 | 6 | 3I2, 1X | Second expiration date M, D, H. |

Format repeats: 6(F5.2, 1X, I4, 3I2, 1X)

The amplitude magnitude correction file may also contain the instrument type (USGS analog short-period or Wood-Anderson, for example). If the instrument type field is left blank, the value set in the station file is used. If the field in the correction file is non-blank, it over-writes the station file value. Use the XMC command to read the XMAG correction file. The choice of which Y2000 format is used is governed by the "200" command. For the instrument type codes, see the "seismometer instrument types" part of the section on amplitude magnitudes. Corrections can also suppress the magnitude weight of individual stations by adding 5.0 (see text above).

Amplitude magnitude correction file format (pre Y2000)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2, 2X	Station net code.
11	3	A3, 1X	Station component code.
15	1	A1, 1X	Instrument type code. Leave blank to use type from station file.
17	5	F5.2, 1X	First magnitude correction.
23	2	I2	Year of expiration of first magnitude correction.
25	6	3I2, 3X	M, D, H expiration date and hour of first magnitude correction. Leave the expiration of the last magnitude correction blank to indicate no expiration. Minutes, if supplied, is ignored.
34	5	F5.2, 1X	Second magnitude correction.
40	2	I2	Second expiration year.
42	6	3I2, 3X	Second expiration date M, D, H.

Format repeats: 6(F5.2, 1X, I2, 3I2, 3X)

Amplitude magnitude correction file format (Y2000 century compatible)

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code.
7	2	A2	Station net code.
9	2	A2	Station location code.
11	3	A3, 1X	Station component code.
15	1	A1, 1X	Instrument type code. Leave blank to use type from station file.
17	5	F5.2, 1X	First magnitude correction.
23	4	I4	Year of expiration of first magnitude correction.
27	6	3I2, 1X	M, D, H expiration date and hour of first magnitude correction. Leave the expiration of the last magnitude correction blank to indicate no expiration.
34	5	F5.2, 1X	Second magnitude correction.

| 40 | 4 | I4 | Second expiration year. |
| 44 | 6 | 3I2, 1X | Second expiration date M, D, H. |

Format repeats: 6(F5.2, 1X, I4, 3I2, 1X)

Other station comments

The relationships used in handling arrival times and delays are as follows:

TOBS = SEC + CCOR - OT

RES = TOBS - TCAL - DLY

where

   SEC = observed arrival time
   CCOR = clock correction (only read from traditional format phase files)
   OT = origin time
   TOBS = observed travel time
   TCAL = calculated travel time
   DLY = station delay
   RES = travel time residual

If a station is found in the phase but not the station file, an error message will normally go to the terminal and print file. This message may be suppressed for a certain list of fictitious stations, such as those referring to digitized time code traces. Use the UNK command to set the list of 5-letter site codes for which you want no error message. All phase data for "unknown" stations is saved in a separate memory area and will be written at the end of the event in the output archive file, but will not be listed in the print file.

PHASE DATA INPUT FORMATS

The name of the input phase data file is specified with the PHS command. The LOC command starts locating events. For example:

```
PHS '1983.PHS'

LOC
```

The phase file may contain any number of earthquakes. Phase data may be in one of several formats (see the COP and FIL commands). All formats require a terminating line after each event. The terminating record may be completely blank, or may contain trial hypocenter data or an ID number. Each station may report any or all of (1) P time, (2) S time, (3) amplitude or (4) coda duration.

An arrival time will <u>not</u> be recognized and processing will continue if any of the following are true:

1) The remark field ("IP" or "ES") is blank. (If the S remark is blank but the S time is non-blank, the reading will be used.)

2) The station is not in the station file.

3) The phase line is incomprehensible or in a bad format.

4) The P arrival time and date differs by more than 6 minutes from the first station (S times are not checked). If you are reading an archive format file and using the earthquake location from the header as a trial location, the P phase times are checked against this reference time.

5) The station site code is on the list of "stations" to ignore given by the UNK command.

Hypoinverse naturally reads the "traditional" phase format (format #1 selected with the COP command), which is compatible with HYPO71 phase lines. It looks for other data in fields left blank or undefined by HYPO71. Hypoinverse archive output may optionally be read back in as input. Another option is the "shadow" format where every input line (phase, archive and terminator lines) are followed by a line with additional data and which begins with a "$". The input shadow record will be copied to the archive output if shadow records are also selected for output. The first line of each event (summary headers with event information) in archive files may have from 1 to 4 shadow lines. The archive format comes in two flavors: older pre-Y2000 files with 2-digit years and newer Y2000 files with 4-digit years including the century. CUSP input is not an ASCII format but invokes a series of database calls that read a binary "MEM" file. CUSP input and output is only supported on VMS platforms.

One can choose the input format explicitly with the COP command, or let Hypoinverse determine the format with the FIL command. To determine the format, set the input file name with the PHS command (which must be an ASCII file, not CUSP binary). Then type FIL, and Hypoinverse will tell you what kind of file it thinks it is, and set the input format accordingly. It will also invoke the Y2000 formats for all input and output if it finds a Y2000-format input archive file. FIL will also set the appropriate archive output format (in place of the CAR command). For convenience, FIL also detects the four different kinds of summary output formats, which are illegal as input formats because they do not have phase times.

Some formats have shadow records, which contain additional data that is not processed by Hypoinverse, but just passed through. Shadow records begin with a "$" character and follow each header, phase or terminator line. In the "archive" formats, the summary header lines may have from 1 to 4 shadow records. If you stripped off all the shadow records, you would have an ordinary archive file. The shadow format in use by the NCSN is given in the output formats section as an example, but Hypoinverse just treats shadow records as ASCII strings to be written to output.

The format choices with the COP command are as follows. The numbers are the value to supply to the COP command. Use the "200" command to invoke either the pre-Y2000 or Y2000 century compatible archive formats with options 3 or 5.

1) "Traditional" Hypo71-style phase format, 2-digit years only. (The default)
2) No longer used.
3) Archive format generated in an earlier Hypoinverse run.
4) Shadow phase format with a 1-line header and a shadow record after each line. (This format is obsolete and is PHASEOUT's HYPO71 shadow option)
5) Archive format with shadow records. (This format is compatible with the PHASEOUT program's Hypoinverse option)
6) Locate one CUSP event, where the CUSP-ID number is given with LOC command.
7) Locate several CUSP events, with ID numbers given in a file.

Archive output format

The archive format is special because it is both an input and output format. Many fields in the archive format are ignored on input because they are filled in with calculated results on output. The archive output format command (CAR) has two choices. Shadow records on input and output may be selected independently of each other. If you select shadow output but not input, an empty shadow record will appear after every output record. Use the "200" command to invoke either the pre-Y2000 or Y2000 century compatible archive formats with formats 1 or 3. Because the "200" flag is a global one, it is not possible to read pre-Y2000 and write Y2000 output.

1) Full archive format. (The default) Corresponds to COP 3 input format.
2) No longer used.
3) Shadow format. Corresponds to COP 5 input format.

Weight codes, name codes, and other parameters common to all phase input formats

Each phase is accompanied by a weight code that states the weight the phase carries in the solution, from full use to no use. By default the codes are:

- 0 or blank = full weight
- 1= _ weight
- 2= half weight
- 3= 1/4 weight
- 4-9 = no weight.

The actual numerical weights of codes 0-3 can be reassigned with the WET command.

The weight value assigned to a P time, S time, duration or amplitude is the product of those specified in the station and phase files.

The station site code must agree exactly (upper/lower case, position of blanks, etc.) with the name in the station file. The site code may not be all-blank. To avoid confusion with shadow lines that could cause some programs to fail, the name should not begin with a "$".

The station component may either be specified in the 1-letter field or the 3-letter field. The 3-letter component in memory is ultimately matched with that in the station file to identify the station (using the number of letters (0-3) chosen with the LET command). If you have chosen 1-letter components (with the LES command), the 1-letter component is transferred to the 3-letter component field before the single letter is matched to the station list in memory. Thus you can use either the 1- or 3-letter fields. If you use the long field for station matching, you can think of the short field as an abbreviation that is easier for humans to recognize.

The traditional Hypo71-style phase format contains an optional 3-letter event remark. For historical reasons, this is abbreviated to 1-letter for output. A 1-letter event remark may be derived from the first 3-letter remark encountered, and output to the summary file. The first 3-letter remark encountered in the input series of phases will be abbreviated to the 1-letter event remarks, if they match one on a preset list. Note that Hypoinverse assigns a second 1-letter event remark if there were problems with the solution such as depth held fixed. In the following _ is a blank and # is any character. The pre-set remark list now consists of: "FLT" & "F__" (becomes F, felt); "TRM" & "T__" (becomes T, tremor associated); "LP#" (becomes L, long period); "BLS" (becomes B, quarry blast); "Q##" or "*##" (becomes Q, quarry blast); or "NTS" (becomes N, NTS shot).

Archive files read as phase input

Formats for the archive files are described later in the section on output files. Many remark fields are simply read in and passed through to output. All of the shadow lines and the terminator line (with any trial hypocenter) are passed through to output without modification. Pass-through fields on the summary line include:

- The first of the two 1-letter auxiliary remarks (the one assigned by the user; quarry blast, etc).
- The externally-determined magnitude, its label and number of readings.
- The event identification number
- The three 1-letter authority and version remarks (see below)

Pass-through remark fields on the station archive lines:

- P & S remarks, first motion, weights, etc.
- 1-letter station seismogram remark
- Data source code (who or where the data came from)

Many fields that are calculated by Hypoinverse are ignored on input, for example:

- 3-letter location remark.
- Station distances and emergence angles.
- Weight values actually used in the final location and importance values.
- Travel time residuals.

If an archive file is read as phase input, the 1-letter event remark assigned by the user is passed through to output. Hypoinverse assigns the second 1-letter remark. A pound sign (#) indicates convergence problems with the final solution such as running out of iterations, or failure of the hypocenter to reach a minimum in RMS. A minus sign (-) indicates the depth was held fixed either by the user or by insufficient data. An (X) indicates the location was fixed to the trial hypocenter.

The three status remark codes for the event found on the input archive header record (and passed through to output) are:

A is the "authority" code, i.e. what network furnished the information. Hypoinverse passes this code through.

V is the "version" of the information, i.e. what stage of processing or completeness the event is in. This can either be passed through, or can be set by Hypoinverse to all events in the current run to the code set with the LAB command.

H is the "version number of last human review". Hypoinverse passes this code through.

The A, V and H codes are in columns 114, 163 and 164 of the Y2000 summary line.

The traditional USGS phase data input format (not Y2000 compatible)

Some fields were added after the original HYPO71 phase format definition.

Start Col.	Len.	Fortran Format	Data
1	4	A4	4-letter station site code. Also see col 78.

5	2	A2	P remark such as "IP". If blank, any P time is ignored.
7	1	A1	P first motion such as U, D, +, -, C, D.
8	1	I1	Assigned P weight code.
9	1	A1	Optional 1-letter station component.
10	10	5I2	Year, month, day, hour and minute.
20	5	F5.2	Second of P arrival.
25	1	1X	Presently unused.
26	6	6X	Reserved remark field. This field is not copied to output files.
32	5	F5.2	Second of S arrival. The S time will be used if this field is non-blank.
37	2	A2, 1X	S remark such as "ES".
40	1	I1	Assigned weight code for S.
41	1	A1, 3X	Data source code. This is copied to the archive output.
45	3	F3.0	Peak-to-peak amplitude in mm on Develocorder viewer screen or paper record.
48	3	F3.2	Optional period in seconds of amplitude read on the seismogram. If blank, use the standard period from station file.
51	1	I1	Amplitude magnitude weight code. Same codes as P & S.
52	3	3X	Amplitude magnitude remark (presently unused).
55	4	I4	Optional event sequence or ID number. This number may bereplaced by an ID number on the terminator line.
59	4	F4.1	Optional calibration factor to use for amplitude magnitudes. If blank, the standard cal factor from the station file is used.
63	3	A3	Optional event remark. Certain event remarks are translated into 1-letter codes to save in output.
66	5	F5.2	Clock correction to be added to both P and S times.
71	1	A1	Station seismogram remark. Unused except as a label on output.
72	4	F4.0	Coda duration in seconds.
76	1	I1	Duration magnitude weight code. Same codes as P & S.
77	1	1X	Reserved.
78	1	A1	Optional 5th letter of station site code.
79	3	A3	Station component code.
82	2	A2	Station network code.
84-85	2	A2	2-letter station location code (component extension).

The terminator line and trial locations (all ASCII formats)

One terminator line must follow each event. If the line is blank, a standard trial hypocenter is used: it is at the standard trial depth beneath the station with the earliest time, at an origin time two seconds before the earliest time. Trial values for any or all of depth, latitude, longitude or origin time may be specified on the terminator line. If a trial value is absent, the standard value is used. To specify a trial origin time, you must supply hour, minute and second. To specify a trial latitude or longitude, you must supply degrees and minutes. A terminator, which is a

HYPO71-style instruction line and is blank except for columns 18-19, is a valid terminator, but the instruction parameter will have no effect.

To fix the depth for one event only, make the trial depth negative. To fix the depth for all events, set the default trial depth negative with the ZTR command.

To fix the entire location (for this event only) to its trial value, put a non-blank character in column 35 of the Hypoinverse terminator line. This might be used for explosions, for example. The origin time will be computed to minimize the average residual. The non-blank character will be copied to the output print file as, for example, the reason for fixing the location.

You may also use the hypocenter on the event header as a trial if you are reading any of the archive formats (COP 3, 4 or 5). You do this by choosing terminator format 3 with the H71 command. COP formats 3 and 5 expect the header in Hypoinverse format, and format 4 expects the header in HYPO71 format. Using the previous earthquake location as a trial hypocenter may not reduce the number of iterations required or speed-up the location run. That is because several iterations may be required before distance and residual weighting are invoked. As iteration proceeds, you may thus find that the hypocenter starts at its trial location, iterates away, then returns to the trial location after all weighting takes the same effect it had to produce the earlier location. On the other hand, a good trial location may help Hypoinverse find the right RMS minimum or put the earthquake in the correct neighborhood if it is unconstrained, and thus produce a better solution.

An optional 10-digit ID number may be supplied in columns 63-72 (a right justified integer) of the terminator line. It will appear in the print, archive and summary outputs.

Columns 1-4 of the terminator line must be blank. Use the H71 command to select either the Hypoinverse or HYPO71 terminator formats, or to get the trial hypocenter from the header if reading an archive format. If all terminator lines are blank, it does not matter which format is used. Either format may be used to enter a trial depth or to fix the depth, but only the Hypoinverse format allows a trial epicenter or origin time on the terminator line.

Hypoinverse Terminator format

Start Col.	Len.	Fortran Format	Data
1	4	A4, 2X	Must be blank.
7	4	2I2	Trial hour and minute.
11	4	F4.2	Trial second.
15	2	F2.0, 1X	Trial latitude (deg).
18	4	F4.2	Trial latitude (min).
22	3	F3.0, 1X	Trial longitude (deg).
26	4	F4.2	Trial longitude (min).
30-34	5	F5.2	Trial depth (a negative value fixes depth).
35	1	A1	A non-blank character fixes the entire hypocenter.
63-72	10	I10	Optional ID number.

HYP071 terminator format

Start Col.	Len.	Fortran Format	Data
1-4	4	A4	Must be blank.
19	1	I1	To fix depth, either put a 1 here or make trial depth negative.
20-24	5	F5.2	Trial depth.
63-72	10	I10	Optional ID number.

Reading earthquakes directly from CUSP "MEM" files

Hypoinverse can locate earthquakes directly from CUSP "MEM" files. There is one MEM file per event whose name contains the CUSP-ID number (i.e. X100229.MEM). You need not be in a CUSP environment or be on a computer running CUSP, but you must be running VMS and define one name before running Hypoinverse: on the Menlo Park VMS cluster, for example, type

```
DEFINE EVENT_DDL PUB1:[CUSP.HYPO]EVENT.DDL
```

The subroutines to handle the MEM file input and output were written by Alan Walter.

Hypoinverse locates events given their CUSP-ID numbers, which are supplied in one of two ways: (1) type in the CUSP-ID numbers and locate events one at a time; or (2) provide a file listing the CUSP-ID numbers of the events to locate. You may use the latter to locate all the MEM files in a directory by making a directory listing to a file before running Hypoinverse (see below).

You can use the existing location in the MEM file as a trial hypocenter by setting the second argument of the H71 command to 3. If you want to examine the data that Hypoinverse is finding in a MEM file, define an output print file and set the print level to a number greater than 6 (like KPR 7). This will generate much output, but is useful for debugging.

There is a substantial list of limitations to the CUSP capability:

- o Station data must be read from an external file as before (using STA or RST).

- o Hypoinverse cannot get instructions from the CUSP scheduler.

- o Hypoinverse must be run in the directory containing the MEM files. As before, all other input and output files set in Hypoinverse may contain pathnames to other directories.

- o At present, locating events from MEM files is noticeably slower than reading ASCII phase files. There is an overhead in making database calls and sifting through unwanted information in the MEM file.

- o CUSP MEM files are several times larger than an equivalent ASCII phase file because they contain seismogram data and entries for traces that were not picked.

CUSP data is treated like another input format selected with the COP command. Use COP 6 to locate one event at a time by CUSP-ID number. In this mode the number is supplied with the LOC command (i.e. LOC 100229). If you just type LOC, you will be prompted for the ID

number. You must already have read in station and crust files, and define other parameters as before. In this single event mode, any phase file specified with the PHS command will be ignored.

You may process a series of events by supplying a file listing their CUSP-ID numbers. Select this CUSP list option with the COP 7 command and specify the file containing the list with the PHS command. The file may only have one CUSP-ID per line, but Hypoinverse will ignore any lines containing a blank field or spurious text. The number should be right-justified in its input field. Use the FID command to specify the format for reading the list file (the default is FID '(I10)'). If you want to locate all of the MEM files in a directory, type the VMS command

```
        DIR/COL=1/OUT=CUSPLIST. *.MEM
```

before running Hypoinverse. In Hypoinverse, use the commands

```
        COP 7
        PHS 'CUSPLIST.'
        FID
```
If all of the CUSP-ID's are 8 digits, use FID '(1X,I8)'. If you have a mixture of 7- and 8-digit numbers, you will have to edit the file to get a field with only right-justified numbers in it. Locate all the events with the LOC command.

It is possible to write the new location back into the MEM file. The COP command takes a second argument if the format choice (the first argument) is 6 or 7. The second argument controls the MEM output destination: 0=none, 1=data structures, 2=shared memory region (if working in a CUSP process), and 3=MEM disk file.

Input of new phase data from the keyboard

If you do not have files of phase data, Hypoinverse has a utility to input arrival time data from the keyboard and write a traditional phase file that can subsequently be located. This utility is entirely separate from the location functions of Hypoinverse. The phase data may come from a reading sheet or other external source. The phase data input utility is invoked with the INP command (no arguments). You must be prepared with a small file containing a list of up to 100 station names (see below). You will be prompted for data from these stations, and will not have to enter station names unless you want to add to the station list as needed. The input utility is mostly self-explanatory and prompts for the data and the decisions it needs. A description of its operation follows.

The input utility first asks for the filename to which phase data will be written. If an existing file is named, any data there is appended to. If a file called "stalist.dat" is present in the current directory, it is read for the list of station names to prompt for on each event. If the file "stalist.dat" is not present, another filename is requested and must be given. Note that the first 4 fields of the station prompting file format are the same as the Hypoinverse station format #2. The prompting list of stations is a convenience: data for other stations may be entered, but the additional station name must be given along with each datum. Also, data need not be entered for every station in the prompting list. Responding with just a RETURN either skips that item, or takes the default value, and goes on to the next piece if information in the list. Stations reporting data for most events should therefore be in the prompt file for the most efficient operation. The "stalist.dat" file also controls whether S-times, amplitudes and durations are routinely requested for certain stations in addition to the P-times that are requested for all stations.

One option is to not enter remarks (such as "IP") and weights (0-4) for phases. If you select this option, all phases get full weight and a simple remark. P or S remarks are 4-letters and include blanks such as "IPU0", "ES_2", or

"_P__" (where _ is blank). Any numerical P or S time, amplitude or duration assigned a zero value is automatically given zero weight (weight code 4). If a P time is in fact 0.00, enter 0.01 instead.

Because INP writes a traditional format phase file, a header line with a location is not required, and a century is not possible. Years must therefore be 2 digits. When locating this file, you will specify the century (with the "200" command) that will be used for all events in the file.

Station prompting file format

Start Col.	Len.	Fortran Format	Data
1	5	A5, 1X	Station site code. A P-time will be requested for all stations.
7	2	A2, 1X	Station net code.
10	1	A1	1-letter station component code.
11	3	A3, 1X	3-letter station component code.
15	1	A1	If non-blank, prompt for an S-time for this station.
16	1	A1	If non-blank, prompt for an amplitude for this station.
17	1	A1	If non-blank, prompt for a duration for this station.

The following prompts and entries are made for each earthquake event you wish to input: (1) Enter the event date and time. For the second and later events, pressing RETURN to a prompt for year, month, day, hour or minute gets the default, which is that of the prior event. (2) Prompts will be made for each station on the prompt list. The P-time is entered first, followed by S-time, duration, and amplitude if you requested prompts for them. If you chose to enter full remarks and weights, prompts for these will follow those for each P and S time. (3) When the pre-set station list is complete, you will be asked if you want to enter another P or S time. If yes, you must enter the full station name as it exists in the station file, time and full remark. Pressing return and taking the default value of 0 skips that item. Basic remarks such as "_P_0" and "_S_0" are provided as defaults. (4) When the event is complete, you will be asked whether you want to stop entering data and return to Hypoinverse command level. If you answer f or false, you will be asked for the date and time of another event, which you must complete.

IMPORTANT NOTES: (1) The case (upper or lower) of station names must agree with that in your station files. (2) All numeric data may be entered in free-format (whole numbers may omit a decimal point, leading zeros are optional, etc.). Alphanumeric input such as station names and remarks, however, is used exactly as entered. (3) Once you press RETURN, the only way to correct mistakes is by leaving the Hypoinverse command level, editing the phase file, and returning to locate the corrected file.

EARTHQUAKE LOCATION METHODS

Interactive earthquake processing

Hypoinverse can locate a set of events interactively. You can alter the input data and relocate several times until you are satisfied with an event. Hypoinverse does this by stepping automatically through a preset list of events you wish to process. The data for each event must be in separate files, and therefore the computer file system does the necessary updating of files and retrieval of the correct events. The two Hypoinverse commands that accomplish this process are BAS to establish the file naming you will use and PRO to actually process (locate and edit) the set of events.

The first step requires putting each event you wish to locate in a separate file. The filenames must consist of a base name of 1-20 characters and a suffix of 1-8 characters for each input and output file type. The same base name is used for each input and output file type associated with

an event, and each event must have a unique base name. I suggest you use the date and time to make up the base name because files will then list in chronological order. The base names are read as a text string from a file listing all events to be processed. For example, a base name might be 19840502133045 and a set of suffixes might be ".arc" (input), ".arc" (output archive), ".prt" (output print) and ".sum" (output summary). The input filename to read for this event would then be 19840502133045.arc. The minimum suffixes required are the input and print files. Note that input and output files can both be ".arc" files with the same name. Hypoinverse reads the input file, closes it, then writes the output file, which would replace the input file on most operating systems.

A good way to build event files is as follows. First, locate an entire set (a month say) so that you have summary and archive files to work with. Then select the events you want to reprocess. You can use a program like SELECT or EQSELECT to get a summary file of events to reprocess, or type the date and time fields (in Hypoinverse summary format) into a file. Then run a program like EXOCET (similar to EXTRACT) to put each selected event into a separate file. The filenames will be of the form YYYYMMDDHHMMSS.ARC (like the example above). After reprocessing, these files will be replaced by their revised versions and can be reassembled into one file (in chronological order) using the VMS "COPY" command or the unix "cat" command. Note that if the origin time of the event changes from the original one during reprocessing, the filenames will always be based on the original origin time. There are two ways to handle the events that will not be reprocessed: If you have a MERGE program, get a file of "rejected" (good) events when you run EXOCET. Later merge the good and repaired files together, and there will be no duplicated events. The simpler way, if you will not use a MERGE program, is to run EXOCET with "none" for a summary file. This will make an individual file for every event. Hypoinverse will reprocess only the selected events, and all events (both reprocessed and original) can be reassembled back into a single file.

The Hypoinverse BAS command defines the filenames to be processed interactively. First specify the file containing base file names of events to process. This may either be a summary file or a list of files produced with a VMS command like DIR/COL=1/OUT=filename *.ARC. Also supply the number of characters in the base name, and the format (like A12) for reading the base name. Any blank spaces in the base names will be filled with zeros. Any line beginning with * or blank will be skipped. Any lines with an invalid filename will generate an error message but will be skipped over.

The BAS command also requests the filename extensions for the input, archive, summary and print files. If you don't want summary or archive files, specify "none" for those names. If your input phase and archive filenames are the same (my preference), Hypoinverse will read back the file it just wrote during interactive processing and all of your edits will be cumulative and saved. If the names are different, you might lose some changes unless they were explicitly made to the input file.

One of the processing steps is the examination of the print output file using an editor. It is best to use a window with 132 columns to see all of the file. You can browse through the file and change the P, S and coda weights for the next location try. You will actually edit the print file to change P, S and coda weights. Any other types of changes must be made to the input phase file directly. If you choose to locate the event again, the print file will be read after the input file so that any changed weights will override the original ones. Note that weights changed in the print

file will be passed to the output archive file. If this archive file is then read as input (input and archive file suffixes the same), the weight changes will be carried along and not lost.

You have a choice of editors. The BAS command asks for your editor choice. The unix version of Hypoinverse will call either dtpad (the solaris window editor), vi or textedit (the sunos window editor). The command you choose should be in your search path. The VMS version lets you choose either EDT or "ED", which you can define how you like.

How to change weights in the print file

Make the change on the line showing the station you wish to change. In identifying stations marked in the print file, Hypoinverse uses the number of letters in the site, net and component codes you selected with the LET command. It also uses the data source code in column 73 of the print file (in case there are multiple sources for the same station). If the station code is blank on the line you change (as for multiple readings from the same station), it will be inferred from the non-blank line above.

- o To change a P weight, put a new weight code in column 1 (before the station site code).
- o To change a coda weight, put a new weight code in column 9 (between the net and component codes).
- o To change an S weight, put a new weight code in column 13 (after the 3-letter component code).

This example shows the columns read for new weight codes in a typical station line:

```
                1111
column: 1234567890123
                c
                o
weight          d
code:   P       a   S
        |       |   |
station V       V   V
line:    BCS  NC VHZ 01V  2.1… (site, net, component, location,
                                 & 1-letter component codes)
```

These are the weight codes currently recognized:

- 0-9: New 1-digit weight code to replace old one.

- \- Remove reading by adding 5 to weight code.

- \+ Restore reading by subtracting 5 from weight code. Existing partial weight codes (1-3) will be converted to full weight (0).

- ! Weight out this and all following P and S weight codes.

Flow of steps in interactive processing

1) Read the base name for a new event and form the input and output filenames. When no base names remain in the file you are done.

2) Open the event input and output files, locate the event, then close the files. If you used the REP command to report events to the terminal, you will see a list of the previous tries for this event with the most recent at the bottom so you can compare them and (hopefully) note improvements.

3) You will then be put in an editor editing the print output file. You may examine the file and change only the P, S and coda weights by putting codes in certain columns. The history of successive location tries is also written to the print file.

4) If you are satisfied with the event, you can QUIT the editor without saving changes to the print file. If you change weights, you must save changes (EXIT in EDT) you made in the editor to keep the changes.

5) You will then get a prompt from the primary branch point within the processing loop in Hypoinverse. The possibilities and their actions are:

- return. This event is OK, go on to next event (step 1 above).

- T   Relocate the current event. Go to step 6 below.

- ZXZ   Delete the current event, including all versions of all input and output files. Then return to step 5.

- KS   Kill all S weights and relocate. Go to step 6 below.

- KA   Kill all P & S weights and continue to the next event. This is useful to preserve the data, but then merge the event with another set of picks for the same event.

- Any other response typed to this prompt will be interpreted as an operating system command. You may do any special operation such as (on VMS) delete the most recent version of the archive file to cancel changes you just made (`DELETE 800101120030.ARC;0`). If you want to stop processing, issue a ctrl-C (unix) or ctrl-Y (VMS) at this point.

6) Relocate the current event. Hypoinverse first asks whether you want to edit the input phase file. This is where you can make <u>any</u> change, such as deleting stations, changing data, flagging quarry shots, inserting a trial hypocenter, etc. When the event is relocated, the input file is read first, then the print file for new weights. Note that weight changes in the print file will override those in the input file, so do not attempt to change weights in the input file. Then go to step 2 above.

When you finish processing a set of events, delete all of the print files. If you are on VMS, purge the old versions of the input, archive and summary files because they will not be needed. If you interrupted the processing session without finishing, you can edit the file listing the base names to either remove or comment-out the events already processed before restarting.

Fixing depth or hypocenter

Hypoinverse offers a few options for fixing hypocentral parameters. One option is fixing the depth while solving for epicenter and origin time. This can be done for all events in a run by

making the trial depth (set with the ZTR command) negative. Fixing depth can also be done on an individual event basis by using a negative trial depth on the terminator line. To specify a trial depth on the terminator you must read an ASCII file. You can't fix a single event depth while reading CUSP MEM files because there is no terminator.

You can fix the hypocenter in one of two ways. The way to fix the entire location (for this event only) to its trial value is to put a non-blank character in column 35 of the Hypoinverse terminator line. This might be used for explosions, for example. This flag to fix the hypocenter will be copied to the output archive file, and thus will remain in effect until it is deleted from the file. The origin time will be computed to minimize the average residual. The non-blank character will be copied to the output print file as, for example, the reason for fixing the location.

To fix the hypocenters for all events in a location run, you prevent any iterations away from the "trial" hypocenter. You do this by setting the maximum number of iterations ITRLIM to zero with the CON command. Be sure to specify a trial hypocenter and origin time when using 0 iterations. Also be sure to invoke the distance and residual weighting on iteration 0 (DIS and RMS commands) because Hypoinverse will always iterate until the distance and residual weighting begin. All calculations including travel times will apply to your trial hypocenter. The origin time may shift slightly from your trial value: Hypoinverse always removes the weighted average station residual from the trial origin time. This minimizes the RMS residual, solves for an origin time and is useful for finding the origin times of known quarry shots, for example. Thus you can never actually fix the origin time in HYPOINVERSE. The calculated travel times are of course independent of the origin time.

Here is a command file fragment to fix hypocenters (the "/" indicates that the remaining arguments, if any, are left unchanged and that a comment can follow:

```
CON 0 /            Set ITRLIM to 0
DIS 0 /            Begin distance weighting immediately
RMS 0 /            Begin residual weighting immediately
COP 5 /            Read shadow format with a summary header
H71 1 3 1 /        Get the trial hypocenter from the header
```

Keeping or eliminating poor earthquakes

Most networks will encounter some poorly recorded earthquakes that do not have enough data to obtain a solution. You may want Hypoinverse to pass all bad events through to output, especially if the output archive file is your permanent catalog record. Or you may want to screen out the poor events, with fewer events output than are input. Two commands control this behavior.

It is hard to predict how your earthquake will fare in Hypoinverse when you have few readings. Four good P arrivals can solve for the four unknowns of the earthquake problem with an "exact" solution under favorable conditions. In some cases, 4 good P or S readings can produce a solution, but of course there can be ambiguities in cases like good P and S from only 2 stations. If there are less than 4 weighted readings, Hypoinverse may attempt a solution with depth held fixed. Using a trial location may help in cases with few readings.

The MIN command sets the minimum number of weighted phases (P or S phases given non-zero weights when they are input) to require before a solution is written to the archive and summary

files. Phases that are later down-weighted because of large distance or residual are still counted in this total of weighted phases. The earthquake will always be written to the print file so you will see the output, even if there are too few readings. MINSTA is 4 by default, but increasing this will screen out the poorly recorded events.

The JUN command can save the data from poor earthquakes by writing them to the archive and summary files even if the final number of phases was not enough to get a good solution. If you set the junk flag to TRUE, you will force a solution of small and poor (junk) events. If the number of stations remaining after distance and residual weighting are applied is less than the minimum number (set with the MIN command), the event would normally abort. Setting the junk flag to TRUE cancels all distance and residual weighting for events that would otherwise abort.

CODA MAGNITUDES

This section discusses coda (duration) magnitudes, but maximum amplitude and external magnitudes (calculated elsewhere and read in) may also be reported for each earthquake. You can select from two coda, two max-amplitude, and the external magnitude to be the preferred magnitude for the earthquake. Each of these magnitude types should have a single letter designator (L for M_L, etc). It is strongly advised to use a letter for each magnitude type and to keep the letter unique. This will avoid confusion about which magnitudes are computed for each station, and which magnitudes appear for the different event magnitudes.

Types of coda magnitudes and how the coda is measured

Hypoinverse offers a choice between the traditional duration (F-P) magnitude M_D (set with the DUR and DUB commands) and the lapse time (tau) magnitude M_T (set with the TAU command; Michaelson, 1987). The choice of which of the duration magnitude types to use is made with the MAG command. Both use the same duration, coda, or (F-P) values on the phase line.

The lapse-time magnitude code adds the P travel time to the duration to get tau. Hypoinverse uses the calculated rather than observed P travel time so that durations can be specified without P times.

The USGS practice is to determine the end of the coda or "F phase" when the coda decays to 10 mm peak-to-peak on the Develocorder viewer, or an equivalent amplitude on an analog or digital signal. This 10 mm Develocorder amplitude (peak-to-peak) corresponds to 0.25 volts of discriminator output (peak-to-peak) or 205 counts of CUSP (12 bit) digitizer output. Equivalently, this coda amplitude corresponds to 0.12 volts 0-to-peak or 0.06 volts average-absolute-value of discriminator output and to a velocity of 1.729×10^{-5} cm/s. See the amplitude magnitude section below for a description of the seismic system and where the amplitude voltage is measured.

For analog channels, the NCSN Earthworm systems are configured to terminate the coda at 0.06 volts average-absolute-value of discriminator output, regardless of the channel's attenuation setting. For digital stations (no VCO and no attenuator setting), the NCSN Earthworm systems are configured to terminate the coda at the average-absolute-value corresponding to a velocity of 1.729 e-5 cm/s. The Earthworm coda termination value (in digital counts) is determined separately for each sensor/digitizer combination by using the sensor motor constant and the

digitizer's sensitivity. In terms of Earthworm parameters, the coda termination value (endcoda-groundmotion) of 1.729 e-5 cm/s is

EWcodaterm * EWsensitivity@15dB * L4Cmotorconstant =

49.14 count *   0.3519 uV/count   * 0.000001 cm/s/uV = 0.00001729 cm/s

Thus a digital velocity channel and a 15db analog velocity channel will report equivalent coda durations. 15 db was chosen as the reference because this is the "default" gain for which no gain correction is made in Hypoinverse processing. This 15 db assumption is also made for coda magnitudes from analog stations with unknown gain.

The Earthworm end-of-coda amplitude is determined for all velocity channels, analog or digital at any gain, such that the resulting coda is equivalent to that measured by traditional (Develocorder) methods. Thus, as long as the Earthworm system is configured carefully as described above, a coda magnitude relation developed for an analog network can work for an Earthworm system recording digital stations and a catalog with both types of stations can be processed together.

For NCSN Earthworm digital stations (no VCO and no attenuator setting), you do not want to make a gain correction to the coda magnitude. You can suppress the gain correction 1) by using a cal factor of 0.0 (unknown gain, defaults to a 15 db attenuation equivalent), 2) by suppressing gain correction for some or all components (see the DUG command), or 3) turn gain corrections off on a station-by-station basis by adding 10.0 to the coda magnitude correction (set in the station file or read with the FMC command). Note that using a cal factor of 0.0 will preclude you from determining amplitude magnitudes for that station.

Coda magnitude options

Duration magnitudes can also use a station gain correction if the gain is known. Gain may be expressed on the station line as either a calibration factor or as an attenuation in db (positive numbers like 12, 18, etc. See the ATN command). All magnitudes are calculated and output to each file to a precision of 0.01. The exception is that magnitudes output to the pre-Y2000 format summary and archive files are to the nearest 0.1.

Weights can be assigned to codas just like they are to arrival times. Weights can also be ignored (even when they are specified), so that either all or named stations with a positive coda time will get a magnitude that will be weighted equally with the others in computing the event magnitude (see the MAG and FMC commands). The event coda magnitude is the weighted median of the station magnitudes. This means that outliers will have less effect than if an average was taken.

Coda magnitudes (but not lapse-time magnitudes) may use a bi-linear relation with different parameters for durations above and below some cutoff value. In addition to several terms in the main duration relationship, other specialized terms using component type (see FCM command) and non-linear distance and depth terms (see DU2 command) can be invoked to calculate Jerry Eaton's (1992) Central California relationship. Component corrections and non-linear distance and depth terms are applied to coda magnitude 1 and the lapse-time magnitude, but not coda magnitude 2.

Selecting coda magnitude types

Hypoinverse can calculate one duration magnitude for each station and two duration magnitudes for each earthquake. A station's magnitude may either be chosen from one of two possible duration (F-P) relationships or from lapse time, but only from one of these. The availability of assigning three magnitude types to two event magnitudes can be complex but it is very flexible. If more than one magnitude type is in use, the magnitude type used for a station depends on its component code. The component codes can be arbitrarily chosen. Several or all components could be used for the magnitude. For example, one could calculate duration (F-P) magnitude for all "ELZ" components and a lapse-time magnitude for all "EHZ" components. The earthquake could then have both a duration magnitude and a lapse-time magnitude. However, you can't calculate a lapse-time magnitude for the EHZ and ELZ components and an F-P (duration)

magnitude for ELZ components because you may only compute one magnitude type for any individual component.

The two event magnitudes (the medians of a set of station magnitudes) are independent, and each may be chosen from duration relation 1, duration relation 2, or the lapse-time relation. If both event magnitudes are lapse-time, for example, the one set of lapse-time constants are used to calculate magnitudes at all stations. The two event magnitudes are medians of different sets of stations defined by their components. For example, you could have FMAG1 determined by all VHZ and VLZ components, and FMAG2 determined by only the VLZ components.

This is probably confusing, so consider the process in two steps: (1) first choose the one or two magnitude types from the three available (duration 1, duration 2, or lapse-time) you want to calculate for each event. (2) Then choose which stations (using their component codes) you want to assign to each of the two magnitudes. Remember that only one magnitude can be computed for each station, but that two event magnitudes can be computed.

(Step 1) First choose the two types of magnitude to be used from the three relationships available. Define the parameters for the first duration relationship with the DUR command, the second duration relationship with the DUB command, and the lapse-time parameters with the TAU command. Then use the MAG command to assign which of these three types to use for the primary coda magnitude, and which for the secondary magnitude. For example, you could calculate only lapse time magnitudes as your primary coda magnitude, or use the duration magnitude 1 as your primary and duration 2 as the secondary coda magnitude.

(Step 2) Next choose which (if any) stations you want to assign to each of the two magnitudes. You do this by selecting the component codes of stations: you can choose no stations (no event magnitude computed), all stations, or list up to 10 different 3-letter codes. The FC1 command selects which components to use for the primary coda magnitude, and the FC2 command selects which components to use for the secondary magnitude. *The default is to use all components for FMAG1 and no stations for FMAG2.* The component selection applies to all 3 of the magnitude types and is independent of it. A magnitude is calculated for each station, but if it is un-weighted or if it is not on the list of components for FMAG1 or FMAG2, it is not used in that event magnitude.

A special case is when both coda duration magnitudes FMAG1 and FMAG2 are selected for a particular component (i.e. a component code appears on both the FC1 and FC2 lists). Because only one magnitude can be computed per station, Hypoinverse chooses the secondary coda magnitude FMAG2 to represent that station. The component list also determines which of the station magnitudes are used in the two event magnitudes. If a station's component is used for both, the magnitude is used both in FMAG1 and FMAG2 for the event. This is true even when the two event magnitudes use different coda parameters. Thus some station FMAG2's can be used in event FMAG1's, but this could be desirable. See below for an example.

If all this seems confusing, it is because Hypoinverse has been asked to serve different networks with different ways of calculating magnitudes, simultaneously.

Coda magnitude examples

A simple example is to use the original CALNET (Lee et al.) relation for all stations to get the event magnitude. The default settings use durations from all stations to compute coda magnitude 1 (DUR command), and compute the event's FMAG1 from them:

```
DUR -0.87 2.0 0 0.0035 0, 5*0, 9999 0
```

The first 5 arguments are coefficients for the constant, log(duration), depth, distance and linear terms in the magnitude vs. duration relation for durations less than FMBRK =9999 seconds. The second 5 terms are the unused terms for the relation for durations longer than FMBRK. The last two arguments are FMBRK and a gain term coefficient (1 to use gain corrections and 0 not to use gains).

A more complicated example is that of the current (2000) NCSN system. NCSN uses Eaton's (1992) M_D relation for the high and low gain 1-hz seismometer network, and Hirshorn and Lindh's M_Z relation for the low gain vertical seismometer network.

Step 1: Use Eaton's (1992) relation for coda magnitude 1. The "1" as the last argument indicates that gain corrections are to be used if they are available:

```
DUR -.81 2.22 0 .0011 0, 5*0, 9999 1
```

Step 2: Add the extra distance and depth terms to coda magnitude 1. These can only be added to coda magnitude 1 (DUR), not 2 (DUB).

```
DU2 .005 40, .0006 350, .014 10
```

Step 3: Add the component corrections to Eaton's coda magnitude. The "3" indicates there are 3 pairs of component codes and their corrections that follow:

```
FCM 3 'VLZ' -.06 'VLE' -.30, 'VLN' -.30
```

Step 4: Use Hirshorn and Lindh's MZ relation for coda magnitude 2:

```
DUB -2.06 2.95 0 .001 0, 5*0, 9999 1
```

Step 5: Now assign the magnitude types to the two event coda magnitudes. This assigns the Eaton relation (DUR) to FMAG1, and the Hirshorn relation (DUB) to FMAG2. The four arguments of the MAG command are:

1) Relation for FMAG1 (1=coda #1, 2=lapse time (tau), 3=coda #2): 1

2) Whether to use assigned coda weights: T for true

3) Relation for FMAG2 (1=coda #1, 2=lapse time (tau), 3=coda #2): 3

4) The default log(Ao) relation to use in amplitude magnitude calculations. 1 means the relation of Eaton (1992) is used for XMAG. Presently there are 5 relations to choose from, but these choices are treated in the section on amplitude magnitudes.

```
MAG 1 T 3 1
```

Step 6: Assign the 1-letter abbreviation for coda magnitude type 1, namely "D" for M_D. The "4" means there are 4 component codes to follow. These codes are for stations for which coda magnitude 1 is to be calculated. The codes represent high-gain 3-component and low-gain vertical 1-sec seismometers.

```
FC1 'D' 4 'VHZ' 'VHE' 'VHN' 'VLZ'
```

Step 7: Assign the one component code for M$_Z$. The relation assigned by the MAG command to FMAG2 is the Hirshorn and Lindh relation. Note that VLZ appears on both the FMAG1 (FC1) and FMAG2 (FC2) lists. See the confusing discussion above. This means that VLZ stations are assigned FMAG2 as a station magnitude, and that it is used in both the primary FMAG1 and secondary FMAG2 event magnitudes. NCSN feels that this makes FMAG1 more stable as an event magnitude.

```
FC2 'Z' 1 'VLZ'
```

Gain corrections to coda magnitudes

The gain-correction methodology for coda magnitudes was developed for the analog, short period seismometers by Jerry Eaton (1970, 1992). A station gain or attenuation correction is made to coda magnitudes if the flag on the DUR or DUB command is set to 1. Additional control over which components and stations are corrected can be made with the DUG and FMC commands. Further control over applying coda gain corrections can be made to individual stations with the coda magnitude correction (FMC command). See the section above "Station magnitude correction files".

  The gain correction is G:

  G = -log (CAL factor/CAL (at 15 db))   where CAL(15 db) = 3.95

  G = 0.05 * attenuation - 0.753

  Log (CAL factor) = -0.05 * attenuation + 1.35

Note that G=0 for a station with 15 db attenuation (CAL of 3.95). Not using a gain correction is equivalent to assuming an attenuation of 15 db, which is close to the 12 or 18 db typical for most analog stations. Either CAL factors or attentions may be supplied in the station file (see the ATN command), or time-dependent attentions or CAL factors may be read from the appropriate history file (see the ATE and CAL commands). If you do not know the station attentions or CAL factors, you should probably assume an "average" attenuation of 15 dB. You may do this in several ways: (1) Use a CAL factor of 3.95 and select the CAL factor option with the ATN command. (2) Leave the CAL factor unknown by using 0 or blank on the station lines and select the CAL factor option with the ATN command, or (3) Do not use a gain (attenuation) correction in the magnitude by setting DMGN=0 with the TAU command or FMGN=0 with the DUR command. Do not directly specify an attenuation of 15 db because attenuations must be a multiple of 6.

Note that duration gain corrections must be enabled both by setting FMGN=1 with the DUR command <u>and</u> by enabling corrections for some or all components with the DUG command. The default setting of the DUG command is to apply duration magnitude corrections to all components, and the default value of FMGN is 0 (DUR command), thus by default no duration gain corrections are used. You can suppress gain corrections for individual stations by adding 10.0 to the coda magnitude correction. This suppression can also vary with time. See the FMC command.

One must be cautious when using both durations and amplitudes from digital stations. If the gain correction is applied to durations from digital stations, you should note that the same cal factor (gain setting) is used for both types of magnitudes, and that the duration gain correction was only derived for analog stations. This led to bogus

duration magnitudes in Menlo Park's tests because the cal factors set for correct amplitude magnitudes did not give appropriate duration gain corrections.

Menlo Park decided to apply duration gain corrections to analog stations, but not to digital stations. Therefore gain corrections were enabled with the DUR and DUB commands. The DUG command, however, is used to enable corrections only for the component codes corresponding to analog stations. If the component codes for analog and digital stations are the same (they are in the SEED convention, for example EHZ), you must selectively suppress corrections for individual stations by adding 10.0 to the coda magnitude correction. This means that the durations from digital stations are assumed to come from a signal equivalent to an analog station running at an attenuator setting of 15 db or a cal factor of 3.95. In practice, the coda termination level in counts for each station is chosen to correspond to 1 cm peak-to-peak (Develocorder viewer) of an analog station at 15 db attenuation. This also means that codas are comparable to those for an old analog station before it became digital. To work out the coda termination level in counts, study the sections on system response in the amplitude magnitude chapter, and keep in mind that duration magnitudes can also be adjusted for certain stations or components with the FMC and FCM commands.

Another specialized way of making gain corrections arose in Menlo Park in 2002. We will convert component codes to SEED compatibility, and thus will loose the easy ability to distinguish analog from digital stations for purposes of deciding whether to make the coda gain correction. We will implement a station-by-station choice of whether to use the gain correction by suppressing it for digital stations. We will use the magnitude correction option of adding 10.0 to the correction as a signal to suppress the gain correction.

Duration magnitude expressions

The complete form of the duration magnitude expression is

$M_D(f-p) = FMA + FMB*log (f-p) + FMF*(f-p) + FMD*D + FMZ*Z + STACOR + FMGN*G$

The FM coefficients are set by the DUR and DUB commands,
f-p is the end-of-coda (F) minus P-time, or duration,
D is the epicentral distance,
Z is the (positive) depth,
STACOR is the duration magnitude correction for the station
G is the gain correction.
S is the slant distance $S^2 = D^2 + Z^2$

An additional component correction may be set with the FCM command. FMGN is normally 0 or 1 and controls whether the gain correction is used.

The magnitude relation may be bi-linear in log (f-p): Hypoinverse uses the values FMA1, FMB1, FMF1, FMD1 and FMZ1 when f-p is less than FMBRK and FMA2, FMB2, FMF2, FMD2 and FMZ2 when f-p is more than FMBRK. If the relation is linear, set FMBRK to 9000 so that FMA2, FMB2, FMF2, FMZ2 and FMD2 are never used.

Some examples of relations in use by the USGS:

Lee et al (1972):

$M_D(f-p) = -0.87 + 2.0*log (f-p) + 0.0035*D + STACOR$

old Hawaii:

$$M_D(f\text{-}p) = -5.2 + 3.89*\log (f\text{-}p) + 0.0037*D + 0.013*Z + STACOR \ (f\text{-}p<210 \ sec)$$

$$M_D(f\text{-}p) = -0.9 + 2.026*\log (f\text{-}p) + 0.0037*D + 0.013*Z + STACOR \ (f\text{-}p>210 \ sec)$$

new Hawaii:

$$M_D(f\text{-}p) = -1.424 + 1.883*\log (f\text{-}p) + 0.00418*(f\text{-}p) + STACOR \ (f\text{-}p<307)$$

$$M_D(f\text{-}p) = -0.267 + 1.917*\log (f\text{-}p) + STACOR \ (f\text{-}p>307)$$

Hirshorn and Lindh (low gain vertical stations):

$$M_D(f\text{-}p) = -0.71 + 2.95*\log (f\text{-}p) + 0.001*Z + STACOR + G$$

Eaton (1992). The additional terms are defined by the DU2 command.

$$M_D(f\text{-}p) = -0.81 + 2.22*\log (f\text{-}p) + 0.0011*D + Stacor + G + Component \ correction$$

$+ 0.005*(D - 40)$	if D < 40 km
$+ 0.0006*(D – 350)$	if D > 350 km
$+ 0.014*(Z – 10)$	if Z > 10 km

Lapse time (tau) magnitude expressions

The supported form of the lapse-time expression is:

$$M_T(tau) = DMA0 + DMA1*\log(tau) + DMA2*\log\_(tau) + DMLIN*tau + DMZ*Z$$

$$+ DMGN*G + STACOR + Component \ correction$$

tau is the lapse time (P travel time + coda duration f-p).
Z is the (positive) depth.
STACOR is the duration magnitude correction from the station line.
G is the gain correction.
A component correction may be set with the FCM command.

The coefficients DM-- are set by the TAU command. The defaults are: DMA0=-1.312, DMA1=2.329, DMA2=0, DMLIN=0.00197, DMZ=0, and DMGN=1.

AMPLITUDE (LOCAL) MAGNITUDES

Local magnitudes from Wood-Anderson seismometers

The method for calculating local magnitudes is modeled after the reading of maximum peak-to-peak amplitudes from a standard Wood-Anderson torsion seismograph. If amplitude is read from an electromagnetic seismometer with velocity output, it is corrected to an equivalent Wood-Anderson response using Jerry Eaton's XMAG formulation (1970, 1992), the seismometer motor constant, and the response curve of the seismometer and recording system. Digital amplitudes are handled also by using the appropriate system gain. Richter's original magnitude formula is:

$$M_L = \log (A_{WA}/2) - \log(A_0) \qquad (1)$$

where A_{WA} is the maximum peak-to-peak amplitude in mm on the paper record, and $-\log(A_0)$ is an attenuation term and is a tabulated function of distance. The division by 2 is because of the peak-to-peak reading. This formula is applied to all stations of Hypoinverse instrument type 0 (Wood-Anderson). The implementation of this formula in Hypoinverse is:

$$M_L = \log(A_{WA} / (2 \times CAL)) + F_1(s) + F_2(d) + XCOR_{COMP} + XCOR_{STA} \qquad (2)$$

where:

CAL is a dimensionless scaling factor assigned to the station. For example, a low gain Wood-Anderson running at a gain of 700 would use a CAL of 700/2080 = 0.3365 to correct the amplitude to the 2080 displacement gain of a normal Wood-Anderson. For a normal Wood-Anderson instrument, CAL should be 1.0.

$F_1(s)$ and $F_2(d)$ are the $\log(Ao)$ distance correction terms that will be discussed below. They can be a function of epicentral distance d, slant distance s, or both.

$XCOR_{COMP}$ is the correction made globally to all components with a given component code. See the XCM command.

$XCOR_{STA}$ is the individual station correction. This is specified for each site and component at that site. Station corrections may be supplied in the station file, or a magnitude correction history file may be specified with the XMC command.

Amplitude magnitude distance corrections

The function $F_1(s) + F_2(d)$ (where d is epicentral distance, and s is the slant distance $s^2 = d^2 + z^2$) represents the $-\log(A_0)$ distance correction. Several choices of this function are available in Hypoinverse. Although all of these numerical functions are approximations to the same relationship in the earth, often a seismic network uses a variety because different studies may have been used to determine different magnitude scales for different stations.

The default relationship choice is specified by the MAG command for components without specific distance corrections. The assignment of distance corrections to specific component codes is made with the LA0 (zero) command.

The distance corrections available are (where the item numbers are those you specify in the MAG and LA0 commands):

1) Eaton (1992)

$F_1 = 0.821*\log(S) + 0.00405*S + 0.955$ for S<185.3 km

$F_1 = 2.55*\log(S)$ for S>185.3

$F_2 = -0.09*\sin(0.07*(D-25))$ only if D<70 km

2) Bakun and Joiner (1984)

$F_1 = \log(S) + 0.003*S + 0.7$

$F_2 = 0$

3) Richter's (1958) approximation

$F_1 = -0.15 + 0.8*\log(S^2)$ for S<200

$F_1 = -3.38 + 1.5*\log(S^2)$ for S>200

4) Nordquist's (BSSA, 1948) UC Berkeley relation

This option computes F_2 by interpolation within a table.

5) P amplitude (presently unused)

Compute F_1 and F_2 from option 1. Then add F_3 as follows:

$F_3 = -0.08 + 0.00942*D$ for D<52

$F_3 = 0.41$ for 52<D<115

$F_3 = 0.812 - 0.0035*D$ for 115<D<280

$F_3 = -0.168$ D>280

Magnitudes (XMAGs) from electromagnetic (velocity) seismometers

Analog data transmission and recording

The method of calculating magnitudes was developed by Jerry Eaton (1969, 1970, 1992) and implemented in the Hypomag program for the USGS network. It is easier to understand the procedure used in Hypoinverse if we discuss his method and the USGS network recording in simple terms.

Visualize the system from ground motion to seismogram like this, with each box having an input and output:
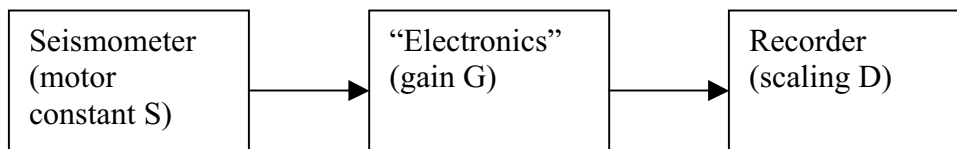
Figure 3. Diagram of simple analog seismic system.

The seismogram amplitude in mm A_D is (and similarly for A_T in digitizer counts)

$A_D = S \times G \times D_D \times H_V$ (3)

For the frequency band in which each system is linear (between the natural period of the seismometer and the high-cut filtering of the electronics, about 3-10 hz), the terms are constants.

- S is the motor constant of the seismometer in v/cm/sec. For the typical L4C seismometer, S = 1.0 v/cm/sec.

- G is the dimensionless gain of the preamp, VCO, transmission and discriminator system. For the USGS system it is $19460 \times 10^{-(atten/20)}$ (or 1228 at 24 db attenuation). The attenuator is in the preamp at the seismometer and is made in 6 db (2x) steps. "Atten" is the setting in db.

- D is the recorder scale factor. For the Develocorder viewer D_D is 40 mm/volt. For the Menlo Park CUSP and earthworm digitizers (also see below) D_T is 4096 counts / 5.0 volt = 819 counts/volt.

- H_V is the ground velocity in cm/sec.

The gain of the electronics and recorder is also expressed as a calibration factor CAL, originally defined as the amplitude in mm on the Develocorder viewer of a 10 microvolt RMS (28.3 microvolt peak-to-peak) 5 hz signal applied to the VCO electronics in place of the seismometer. Although defined historically, CAL is the measure of system gain that Hypoinverse uses because the USGS keeps track of units equivalent to CAL factors as measures of gain.

It is useful to express CAL as a number related to gain G or to attenuation. This assumes the values of D given above and removes them from the definition of CAL. CAL thus does not depend on the recording system used. Simple algebra yields:

$$CAL = G / 883 \qquad\qquad\qquad (4)$$

$$\log(CAL) = 1.35 - 0.05 \text{ x atten} \qquad\qquad\qquad (5)$$

The gain expressed as CAL may be specified for all seismometer types, including the Wood-Anderson and various digital systems. It is only for the USGS-NCSN analog transmission system and networks using that system that the concept of the attenuator setting makes sense. Hypoinverse allows you to specify gain (and gain history) as attenuation only for the analog system using L4C seismometers and instrument type 1 (see below). If the CAL value is set to zero, no amplitude magnitudes will be computed for this station.

Amplitude magnitude relation for velocity seismometers

The "X" magnitude formula developed by Jerry Eaton for velocity seismometers is used in Hypoinverse. Before the magnitude is calculated, the amplitude is converted to an effective Wood-Anderson amplitude using the period at which the amplitude is measured and the response curve for the seismograph type. The magnitude calculation of course assumes that the location of the maximum amplitude on the seismogram is at the same time for both the velocity seismogram and a Wood-Anderson (displacement) instrument. This means the seismogram fits the definition of local magnitude where the W-A seismogram has its maximum amplitude. The magnitude relation is an extension of equation (2):

$$M_X = \log (A_D / (2 \text{ x } CAL \text{ x } R(f) \text{ x } S)) + F_1(s) + F_2(d) + XCOR_{COMP} + XCOR_{STA} \quad (6)$$

- A_D is the peak-to-peak amplitude on the Develocorder viewer in mm.

- CAL is the dimensionless calibration factor depending on system gain as described above.

- R(f) is the frequency dependent response curve of the USGS system relative to the Wood-Anderson seismometer. Hypoinverse interpolates this function from the table given below and plotted in figure 4. In the band 3 hz < f < 10 hz where the seismometer is flat to velocity, the Wood-Anderson is flat to displacement, and no electronic filtering is in effect, $\log(R) \sim \log(f) + 1$.

- S is the seismometer motor constant in volt/cm/sec.

- $F_1(s)$ and $F_2(d)$ are the $\log(A_0)$ distance correction terms discussed above. They can be a function of epicentral distance d, slant distance s, or both.

- $XCOR_{COMP}$ is the correction made globally to all components with a given component code. See the XCM command.

- $XCOR_{STA}$ is the individual station correction. This is specified for each site and component at that site. Station corrections may be supplied in the station file, or a magnitude correction history file may be specified with the XMC command.

Response function R(f) of NCSN analog system relative to Wood-Anderson response

Figure showing log(R) on the y-axis (0.0 to 2.0) versus log(frequency) on the x-axis (-1.0 to 1.5). The curve rises from about 0.3 at log(frequency) = -0.75 to a peak of about 2.15 near log(frequency) = 1.2, then drops steeply to about 0.8 at log(frequency) near 1.7.
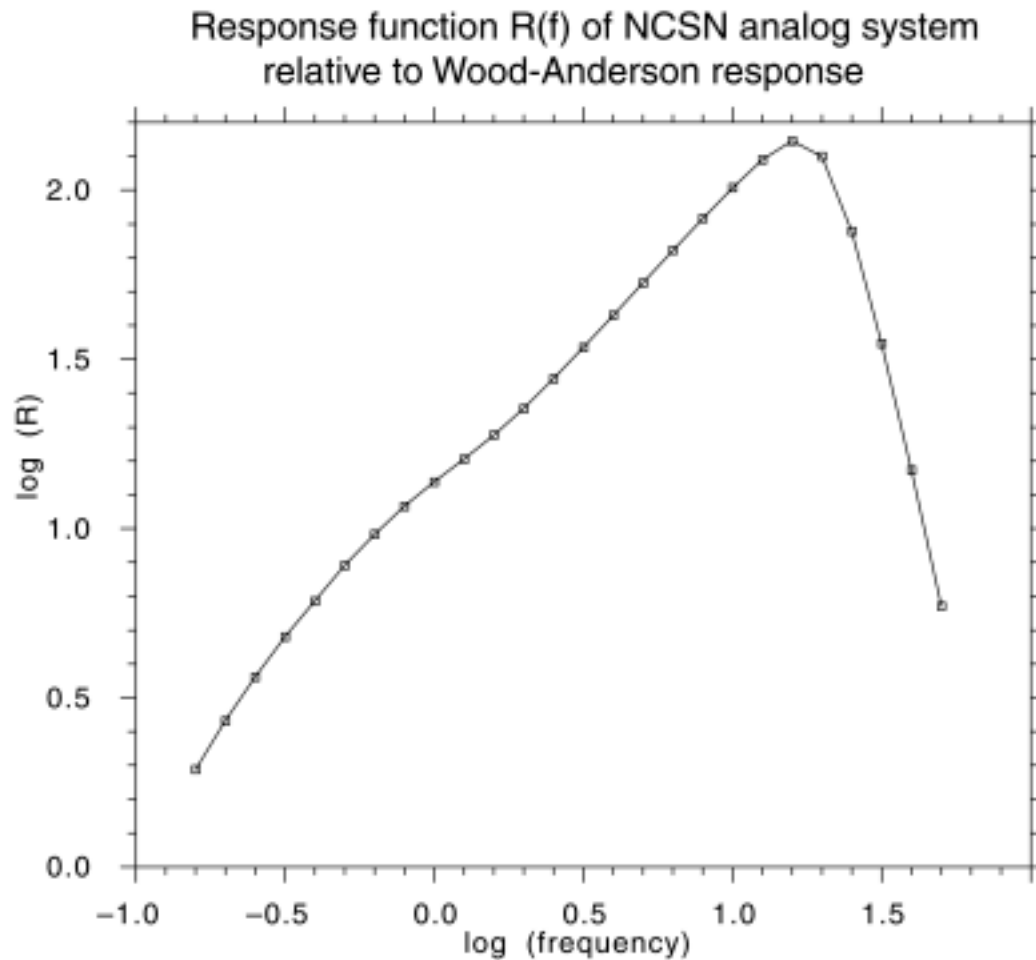
Figure 4. Response function R(f) of the standard NCSN analog seismic system relative to the Wood-Anderson displacement seismometer.

The analog system has a 1 sec. velocity seismometer and high-frequency cutoff filtering. Dividing by this response curve converts velocity amplitudes to Wood-Anderson amplitudes and produces "x" (local) magnitudes.

Hypoinverse uses the function R(f) for all velocity seismometers. These are instrument types 1 and 3-7. The tabulated R(f) includes the response of a 1 hz 0.8 critically damped velocity seismometer relative to a Wood-Anderson, high cut and low cut electronic filtering, and the Develocorder galvanometer response which attenuates high frequencies. The function R(f) is used in the magnitude formula for all velocity response seismometers, whether used with analog or digital recording. As long as you are in a frequency band 3 hz < f < 10 hz where the responses are linear and unfiltered, R should be a good approximation to most systems. Scaling the response to different gains can be done with the CAL factor. The Hypoinverse user could code a new response table if modeling a specific system outside this band is necessary. Here are the R values used by Hypoinverse:

frequency	0.16	0.20	0.25	0.32	0.40	0.50	0.63	0.79	1.00
log frequency	-0.8	-0.7	-0.6	-0.5	-0.4	-0.3	-0.2	-0.1	0.0
log R	0.288	0.432	0.561	0.680	0.786	0.891	0.983	1.066	1.138

frequency	1.26	1.59	2.00	2.51	3.16	3.98	5.01	6.31	7.94
log frequency	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
log R	1.205	1.276	1.355	1.443	1.535	1.630	1.726	1.822	1.916

frequency	10.0	12.6	15.9	20.0	25.1	31.6	39.8	50.1
log frequency	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7
log R	1.921	1.998	2.010	1.918	1.740	1.502	1.196	0.789

Relating M_X to M_L

We can check our equations by equating M_X to M_L and seeing if the velocity seismometer response is reasonable. Equating (1) and (6) yields

$$A_{WA} / 2 = A_D / (2 \times CAL \times R(f) \times S)$$

Using (3) and the relation between H_D in mm and H_V in cm/sec

$H_V = 2 \pi f \, 0.1 \, H_D$ and using

$CAL = G / 883$ and using

$D_D = 40$ mm/volt   yields

$A_{WA} = 22200 \, f \, H_D / R(f)$

If we restrict the frequency band to 3 hz and higher where the Wood-Anderson is flat to displacement and its gain is 2080:

$A_{WA} = 2080 \, H_D$ yields

$R(f) \sim 10.67 \, f$

This is close to the approximation given in the definition of R(f) above.

Digital data recording and units of amplitude measurement

Before the 2000 version, Hypoinverse always assumed that the units of measurement were in millimeters as read from paper records or from the Develocorder viewer screen (100x magnification from the film). The advent of digital seismographs and the reporting of amplitudes in digital counts meant that amplitudes had to be converted to mm before writing on the phase line for input to Hypoinverse. Refer to equation (3). For the USGS analog recording, the scale factor D_D = 40 mm/volt applied to the Develocorder viewer. For Menlo Park CUSP and Earthworm digitizing and recording systems using 12-bit digitizers, 5.0 volts of discriminator output would produce 4096 digital counts. Thus D_T = 819 counts/volt. For the Hawaiian Volcano Observatory CUSP digitizing and recording using 14-bits of a 16-bit digitizer, 5.0 volts of discriminator output would produce 16,384 digital counts. Thus D_H = 3276.8 counts/volt.

The 2000 version of Hypoinverse supports amplitude measurement and input in digital counts in addition to mm. The code for the type of amplitude units being used is located on the Y2000 archive phase format and is a 2-digit integer code. The units code field is next to the amplitude field, which has been expanded from 3 columns to 7 columns. The print file lists the amplitude in the input units, with a letter indicating the type of units. Presently only four amplitude-unit codes are supported:

Table 1. Codes for seismogram amplitude units.

Units code	Amplitude units	U factor	Units code (print file)
0	peak-to-peak mm	1.0	M
1	0-to-peak mm for Berkeley Wood-Anderson data	2	M
2	peak-to-peak digital counts from Menlo Park CUSP or earthworm digitizers	0.04883	C
3	peak-to-peak digital counts from HVO-CUSP digitizers	0.012207	D

Hypoinverse converts amplitudes in digital counts to mm before applying equation (5) to calculate the magnitude. The conversion factors for counts are

$$U_T = D_D / D_T = 0.04883 \text{ counts/mm}$$

$$U_H = D_D / D_H = 0.012207 \text{ counts/mm}$$

The modified magnitude relation used by Hypoinverse for amplitudes in counts is then:

$$M_X = \log((A_C \times U) / (2 \times CAL \times R(f) \times S)) + F_1(s) + F_2(d) + XCOR_{COMP} + XCOR_{STA} \quad (7)$$

where A_C is the peak-to-peak amplitude in digital counts.

Full digital systems with velocity seismometers

This system is similar to that described above, except that the amplifying electronics and the "recorder" can be visualized as combined into a digitizer:
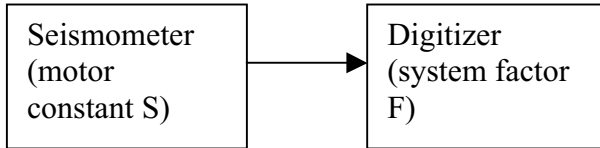
Figure 5. Diagram of simple digital seismic system.

The response of this system is similar to equation (3), where A_C is the output amplitude in counts:

$$A_C = (S \times 10^6 / F) \times H_V \tag{8}$$

- S is the motor constant of the seismometer in v/cm/sec. For the typical L4C seismometer, S = 1.0 v/cm/sec.

- F is the system factor of the amplifier and digitizer in microvolts per count. The 10^6 factor converts from volts to microvolts.

- H_V is the ground velocity in cm/sec.

Combining equations (3) and (8) and the relations between counts and mm of seismogram amplitude yields:

$$10^6 / F = 819 \ G \ \text{counts/volt}$$

$$CAL = 1.382 / F \qquad \text{(Menlo Park and earthworm)} \tag{9}$$

$$CAL = 0.3456 / F \qquad \text{(HVO CUSP)}$$

Thus when you know the system factor F of your system in microvolts per count, you can assign a CAL to that station to compute magnitudes from equation (7) providing the code specifying the type of amplitude units is correct.

The relation (9) between F and CAL assumes that F is used to specify the entire electronic and recording system. It works for a purely digital system. It also works for a hybrid analog-digital system like earthworm or the Menlo Park CUSP system where the digitizer factor is $D_T = 819$ counts/volt and $U_T = 0.04883$ counts/mm.

If the system is like the Hawaiian Volcano Observatory's with a different digitizer factor, the CAL values may not match those of an older analog system. For systems with a different digitizer factor where CAL must match earlier values, specify a different digital count units code in the input phase archive than the value of 2 used for Earthworm or the Menlo Park CUSP systems. Additional U factors can be programmed into Hypoinverse in the future.

If the Hypoinverse input is from a MEM file (COP 6 or 7), the amplitude units code is sensed from the digitizer device code stored in the MEM file. All amplitudes from AMF tuples (amplitudes measured by analyst) are in digital counts. By default the units code is 2, unless the event is from the Hawaiian Volcano Observatory with digitizer device code OBI or XOB, where the amplitude units code will be 3.

Equations (7) and (9) mean that you can compute magnitudes from a variety of velocity sensor and recording systems. The sensor output S (in volt/cm/sec) and system factor F (in microvolts per count) completely specify a system for the frequency band where response in linear (ie. 3-10 hz). You set S by choosing a standard seismometer instrument type, and set F by explicitly

entering a CAL factor. In fact, only the product CAL x S is used for scaling in the magnitude calculation.

Seismometer instrument types

The instrument type code can be specified in the station file or in the amplitude magnitude correction file read with the XMC command. If the type for a station and component is specified in the correction file (with a non-blank entry), it overrides the one given in the station file.

Types 0 and 2 invoke the M_L magnitudes. Types 1 and 3-7 invoke M_X magnitudes for velocity seismometers. Choosing the instrument type also invokes the seismometer motor constant S in the M_X relation. S is in volts/cm/sec.

Hypoinverse does not have a magnitude relationship for accelerometers. Relating acceleration to displacement (Wood-Anderson response) for an estimate of earthquake source energy is a bit of a stretch. In any case, the seismogram phase of maximum displacement typically does not coincide with the phase of maximum acceleration. For non-typical seismometers it is better to convolve the seismogram to Wood-Anderson response before measuring the maximum displacement phase.

Table 2. Seismometer instrument types.

Type	S	Instrument
0	-	Wood-Anderson torsion seismograph
1	1.0	1 hz L4C velocity transducer with 0.8 critical damping. Gain information, if specified as a history file, is stored as **attenuation** history.
2	-	Hawaii type Sprengnether seismometer with optical recording
3	1.0	1 hz L4C velocity transducer with 0.8 critical damping. Gain information, if specified as a history file, is stored as **CAL factor** history.
4	0.448	HS1 (ie. Nanometrics)
5	8.0	Guralp (ie. broadband Reftek)
6	24.0	Streckheisen STS-1 (ie. BDSN broadband)
7	15.0	Streckheisen STS-2 (ie. BDSN broadband)

Types 1 and 3 are for the same seismometer and have the same response. In this case, Hypoinverse also uses the instrument type to decide whether a station's gain history is in the attenuation history file (type 1) or a calibration history file (types 3-7). When processing encounters an expired gain value, Hypoinverse then knows which of the two files to retrieve the new gain value from by checking the instrument type.

For types 1 and 3-7, Hypoinverse uses interpolation within a digitized response curve to find R(f). This is the response curve of an L4C 1-sec. seismometer with analog recording relative to the Wood Anderson. The response R(f) approximates that of all velocity sensors in the frequency band where the response is linear, i.e. 3-10 hz.

For instrument type 2 in the period range 0.1 to 1.9 second, log (R) is approximately a linear function of the logarithm of period:

for type 2: $\log (1/R) = 0.41 - 0.56 \log (0.2/PER)$

CAL factors for various digitizing systems

Knowing the system factor F in microvolts per count will let a network operator assign a CAL factor to a station using equation (9). We give some typical values here for systems in use by the USGS in Menlo Park. The USGS station codes are SEED (Standard for earthquake information exchange) compatible for broadband stations, and were devised within the USGS for the variety of equipment for the short period stations. The dash symbol in the station codes column stands for the direction letter Z, E or N.

The relation (9) as used below between F and CAL assumes that F is used to specify the entire electronic and recording system. It works for a purely digital system. It also works for a hybrid analog-digital system like earthworm or the Menlo Park CUSP system where the digitizer factor is $D_T = 819$ counts/volt and $U_T = 0.04883$ counts/mm. If the system is like the Hawaiian Volcano Observatory's with a different digitizer factor, the CAL values may not match those of an older analog system. For systems with a different digitizer factor where CAL must match earlier values, specify a different digital count units code in the input phase archive than the value 2 used for earthworm or the Menlo Park CUSP systems.

Table 3. Gain factors of different seismic systems.

Seismic system	F (microvolts / count)	CAL factor	USGS station codes	Typical seismometer
NCSN analog, CUSP or Earthworm digital recording (12 bit)	0.9954 at 24 db atten.	1.4 at 24 db atten.	VH-, VL-	L4C
NCSN DST (Digital seismic telemetry, 12 bit)	6.865	0.2014	VD-	L4C
Nanometrics HRD (Hayward boreholes)	1.037	1.333	VD-,EP	HS1, L4C
Nanometrics HRD (broadband net)	1.275	1.084	BH-	Guralp
Reftek broadband net (discontinued)	1.907	0.7252	BH-	Guralp
Quanterra datalogger, BDSN short period	2.5	0.5532	EP1	L4C
Quanterra datalogger, BDSN broadband	2.5	0.5532	BH-	Streckheisen
Kinemetrics K2	0.298	4.637	EHZ	L4C

Amplitude magnitude comments

If the calibration factor CAL is found equal to 0, no magnitude will be computed for that station. The useful range of XCOR is +/-2.4. If you want to compute a magnitude for a station but exclude the result from the event magnitude, either give it a zero weight or use a value of XCOR equal to 5.0 plus the actual correction. See the sections on station and phase input for more information.

If you are calculating amplitude magnitudes from vertical instruments, you should correct magnitudes to account for the lower amplitudes on verticals than on horizontals for which the ML scale was originally defined. The amplitude magnitude correction for vertical instruments in Central California is empirically about +0.25. You could add this to all magnitude corrections of vertical stations in the station file. More easily, if you have unique component codes for vertical stations, a global correction applied to all vertical components may be set with the XCM command.

Computing two amplitude magnitudes

Like coda magnitudes, the philosophy for calculating two magnitudes for amplitude is complicated. Hypoinverse can calculate one amplitude magnitude for each station and two amplitude magnitudes for each earthquake. The Magnitude choice (primary or secondary) used for a station depends either on its component code, or its seismometer instrument type. The choice is made with the XCH command. Several or all components could be used for each magnitude. One could calculate a local magnitude for all Wood-Anderson components and an "X" magnitude for all vertical components, for example. Similarly, one could use the instrument type to select the two magnitudes. The two event magnitudes (the medians of a set of station magnitudes) are independent, but both are drawn from the set of station magnitudes according to the component list (or instrument list) chosen for each magnitude.

Choose which (if any) stations you want to assign to each of the two magnitudes. The first way to assign a station to either a primary or secondary magnitude is by selecting the component codes of stations: you can choose no stations (no event magnitude computed), all stations, or list up to 10 different 3-letter component codes. The XC1 command selects which components to use for the primary amplitude magnitude, and the XC2 command selects which components to use for the secondary magnitude. The default is to use all components for XMAG1 and no stations for XMAG2. A magnitude is calculated for each station, but if it is un-weighted or if it is not on the list of components for XMAG1 or XMAG2, it is not used in that event magnitude. The XC1 and XC2 commands also assign a 1-letter label to each magnitude, a practice that is strongly advised.

If you are using instrument types to choose the event magnitude, use the XTY command. You can't use <u>both</u> component and instrument to choose the magnitude. The method you choose will depend on how your network is set up in terms of the naming of components or the mix of seismometers.

A special case is when both amplitude magnitudes XMAG1 and XMAG2 are selected for a particular component (ie. a component code appears on both the XC1 and XC2 lists). Because only one magnitude can be computed per station, Hypoinverse chooses the secondary amplitude magnitude XMAG2 to represent that station. The component list also determines which of the station magnitudes are used in the two event magnitudes. If a station's component is used for both, the magnitude is used both in XMAG1 and XMAG2 for the event.

P-amplitude magnitudes

Hypoinverse presently does not support P-amplitude magnitudes. Theoretically, P-amplitudes offer a way to estimate magnitude very early in the seismogram before waiting for the coda to run out and determine the duration time. P-magnitudes may be an advantage only for networks with high dynamic range, high frequency, vertical seismometers that do not record S very well. The increase in numbers of broad-band seismometers make reliance on P-magnitudes unlikely. Most of the source code is present within the program, but it is not fully developed and there is no place in the data formats for p-amplitudes.

Amplitude magnitude command example

An example is that of the current (2000) NCSN system. NCSN uses Eaton's (1992) M_X relation for the high and low gain 1-hz seismometer network, and local magnitudes for any of the UC Berkeley Wood-Anderson (or broad band with a WA response) stations. All of these commands are more fully explained below in the command dictionary.

Assign the 1-letter abbreviation for the primary amplitude magnitude, namely "X" for M_X. The "7" means there are 7 component codes to follow. These codes are for stations of instrument type 1 or 3 for which X magnitudes are to be calculated. The codes represent high- and low-gain vertical 1-sec seismometers.

```
XC1 'X' 7 'VHZ' 'VLZ' 'VLE' 'VLN' 'VDZ' 'VDN' 'VDE'
```

Assign the component codes for local magnitudes M_L from Wood Anderson response stations. These are of instrument type 0. L is the magnitude label and 6 means there are 6 component codes to follow.

```
XC2 'L' 6 'WLN' 'WLE' 'HHN' 'HHE' 'BHN' 'BHE'
```

NCSN sets the default log(A_0) relation to use in amplitude magnitude calculations to be that of Eaton (1992). This is indicated by 1 as the fourth argument in the MAG command:

```
MAG 1 T 3 1
```

NCSN uses Berkeley's Nordquist log(A_0) relation (choice 4) for the 6 WA & synthetic WA components:

```
LA0 6 'WLN' 4, 'WLE' 4, 'BHN' 4, 'BHE' 4, 'HHN' 4, 'HHE' 4
```

Add the component corrections to Eaton's amplitude magnitude. The "2" indicates there are 2 pairs of component codes and their corrections to follow:

```
XCM 2 'VHZ'  .33, 'VLZ'  .20
```

The weighting of duration and amplitude magnitudes

The summary duration or amplitude magnitude for an event (as reported in the event header and summary line) is the weighted median of the station magnitudes. The weighted median is the magnitude value for which half of the total weights are higher and half are lower. Amplitude and duration magnitudes are calculated and reported separately and are never mixed.

Separate amplitude and duration weights may be specified on both the station and phase lines. The weight used is the product of the two. The weight codes are the same as the default weights for individual P and S phase times: 0 or blank = full weight, 1= _ weight, 2= 1/2 weight, 3= 1/4 weight, 4-9 = no weight. Specifying a magnitude correction larger than 2.5 will also give that station zero weight: the correction used is the supplied value minus 5.0. Another way to avoid using a whole class of stations in the event magnitude is to omit their component code from those selected for use by the FC1, FC2, XC1 and XC2 commands.

The outputs of Hypoinverse report the weighted median magnitude, the total of all weights (essentially the number of magnitudes used in the average), and the mean-absolute-difference (MAD error) for all four of the amplitude and duration magnitudes.

PREFERRED EARTHQUAKE MAGNITUDES

Hypoinverse potentially has available 5 different magnitudes:
1) Primary duration magnitude

2) Primary amplitude magnitude
3) An externally derived (sometimes called "Berkeley") magnitude
4) Secondary amplitude magnitude
5) Secondary duration magnitude
6) and 7) are reserved for un-implemented P-amplitude magnitudes

Hypoinverse can select a "preferred" magnitude from those available. The preferred magnitude is recorded along with all the other magnitudes, but is generally the one that is reported or plotted. The choice can require that certain criteria are met, have a preference order, and thus be somewhat complicated. The procedure for choosing a preferred magnitude involves going sequentially through your list of choices, and if a given magnitude matches your criteria, it is chosen as the preferred value. The tests made on each magnitude to evaluate it as the preferred magnitude are:

1) It is defined (greater than 0)
2) It have a minimum number of readings (total of weights)
3) It be in a certain magnitude range

For example, NCSN earthquake processing can have any of 4 magnitudes available. The choices are:

1) External magnitude (Berkeley Wood-Anderson), any number of readings (ie. minimum 0), $4.0 \leq M < 9.9$
2) Primary duration magnitude, at least 1 reading, any M ($0.0 < M < 9.9$)
3) Primary amplitude magnitude (xmag), at least 1 reading, any M ($0.0 < M < 9.9$)
4) Secondary amplitude magnitude (recomputed Wood-Anderson), at least 4 readings, $4.0 \leq M < 9.9$
5) External magnitude, any number of readings, any M: $0.0 < M < 9.9$
6) Secondary amplitude magnitude (recomputed Wood-Anderson), any number of readings, any M: $0.0 < M < 9.9$

Use the PRE command to specify the preference order. For each magnitude you want to consider, specify its type number (1-5), the minimum number of readings, and the minimum and magnitude. This NCSN example uses the PRE command:

```
PRE 6, 3 0 4 9, 1 1 0 9, 2 1 0 9, 4 4 4 9, 3 0 0 9, 4 0 0 9
```

WEIGHTING OF P & S TIMES

The actual weight given a P or S time is the product of several factors:

1) The station weight. A code on the station line results in a weight factor between 1 and 0 in increments of 0.25 for that station for the entire run.

2)   The global S weight. This number is set with the SWT command for an entire run and that factor multiplies all S weights. For example, setting SWT to 0.5 gives all S times half the weight they are individually assigned.

3)   The weight assigned each phase. The weight codes 0-4 and blank yield weights from 1.0 to 0.0. The actual weight factors for codes 0-3 may be reset from their default 0.25 increment values using the WET command. The codes 4-9 yield no weight.

4)   Distance weight. Weight decreases from 1.0 to 0.0 with increasing distance.

5)   Residual weight. Weight decreases from 1.0 to 0.0 with increasing absolute value of travel time residual.

The individual P & S weight codes as originally assigned and the final, normalized weight used (to two decimal places) are preserved in the archive and print output files. Note that the weights are normalized (average value of the weight squared is 1.0) for the inversion, and that a fully-weighted station will often have a value greater than 1.0.

Distance Weighting

An ideal distance-weighting scheme should allow for both reducing the weights of distant stations when an event is in the interior of the network, and the maximum use of all stations when an event is outside the network. The Hypoinverse distance weighting function is 1.0 for near stations, 0.0 for far stations, and follows a cosine taper in between. The distance points at which weight tapering begin and end can be made to stretch out and scale with the distance to the second closest station DMIN2 for earthquakes outside the network. The four parameters that govern distance weighting are set with the DIS command and are (with some sample values), ITRDIS, the iteration number at which distance weighting is to begin (4), DISCUT (50 km), DISW1 (1.) and DISW2 (3.). When the second-closest station distance DMIN2 is larger than DISCUT, the distance at which the taper begins and ends scales with DMIN2, and is full weight closer than DMIN2* DISW1 and no weight farther than DMIN2* DISW2. When the second-closest station distance is less than DISCUT, the distance at which the taper begins and ends is fixed, and is full weight closer than DISCUT* DISW1 and no weight farther than DISCUT* DISW2. See Figure 6. To keep the taper distance fixed at, for example, XNEAR and XFAR, set DISCUT = 1000, DISW1 = XNEAR/1000 and DISW2 = XFAR/1000.

1.0

WEIGHT

0.0

DMIN2 > DISCUT
(EVENT OUTSIDE NET)

DISTANCE

(DMIN2·DISW1)   (DMIN2·DISW2)

1.0

WEIGHT

0.0

COSINE TAPER

DMIN2 < DISCUT
(EVENT WITHIN NET)

DISTANCE

(DISCUT·DISW1) (DISCUT·DISW2)
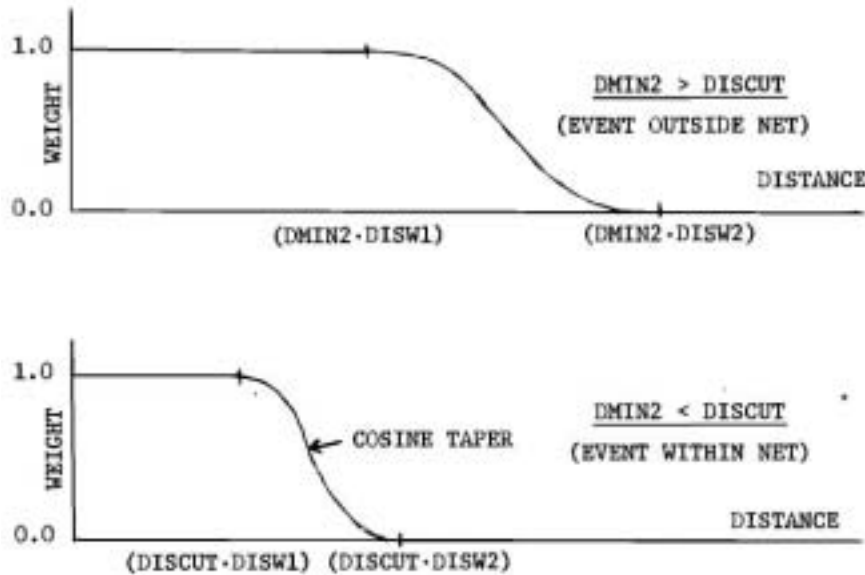
Figure 6. The distance weighting function.

DISCUT, DISW1 and DISW2 are constants set with the DIS command. DMIN2 is the distance to the second closest station. If DMIN2 is larger than DISCUT (upper figure), the function stretches out and scales with DMIN2 such that most of the network stations receive weights somewhere in the tapering part of the function. If DMIN2 is smaller than DISCUT (lower figure), the function is fixed. Stations farther than DISCUT x DISW2 then receive no weight.

Residual weighting

The purpose of residual weighting is to reduce the weight of arrivals with large residuals which may reflect large timing errors or travel paths for which the velocity model is very poor. The residual weighting function is 1.0 for small residuals, 0.0 for large residuals, and follows a cosine taper in between. In Hypoinverse, the residual points at which weight tapering begin and end can be made to stretch out and scale with the root-mean-square residual (RMS). The four parameters which govern residual weighting are set with the RMS command and are (with sample values), ITRRES, the iteration number at which residual weighting is to begin (4), RMSCUT (0.10 sec), RMSW1 (1.0), and RMSW2 (3.0). The residuals at which the taper begins and ends scale with RMS when it is larger than RMSCUT, and are fixed when it is less than RMSCUT (see figure 7). When RMS is larger than RMSCUT, the residual at which the taper begins and ends scales with RMS, and is full weight closer than RMS*RMSW1 and no weight farther than RMS*RMSW2. When RMS is less than RMSCUT, the residual at which the taper begins and ends is fixed, and is full weight closer than RMSCUT*RMSW1 and no weight farther than RMSCUT*RMSW2.

Thus, most stations will be fully weighted when RMS is large, but as iteration and convergence proceed and RMS becomes smaller, the weights of large residuals will lower. RMSCUT prevents an inward spiral in which large residuals are discarded, which then lowers RMS, and which results in more large residuals being discarded, etc. There are two distinct values of RMS used by the program. RMSWT is the RMS residual computed *before* residual weights are applied and on which residual weights are based. The output variable labeled RMS is the value of RMS computed *after* residual weights are applied, and is used for convergence tests (see

below) and as a final quality measure. Residual weighting is re-applied on every iteration (starting with the iteration where it is to begin). This allows stations to be fully weighted for the first few iterations until the hypocenter approaches its final location, and after residual weighting begins, stations may be weighted in or out as convergence proceeds. This generally allows about the largest 10% of the residuals to be weighted down if there are any bad readings. If you have trouble converging, I have found that manually weighting out the largest residuals, then re-introducing them if the residuals become small, often helps in tracking down the bad readings.

To not use any residual weighting, set RMSCUT to a large number like 1000. This forces full weighting of residuals less than thousands of seconds.
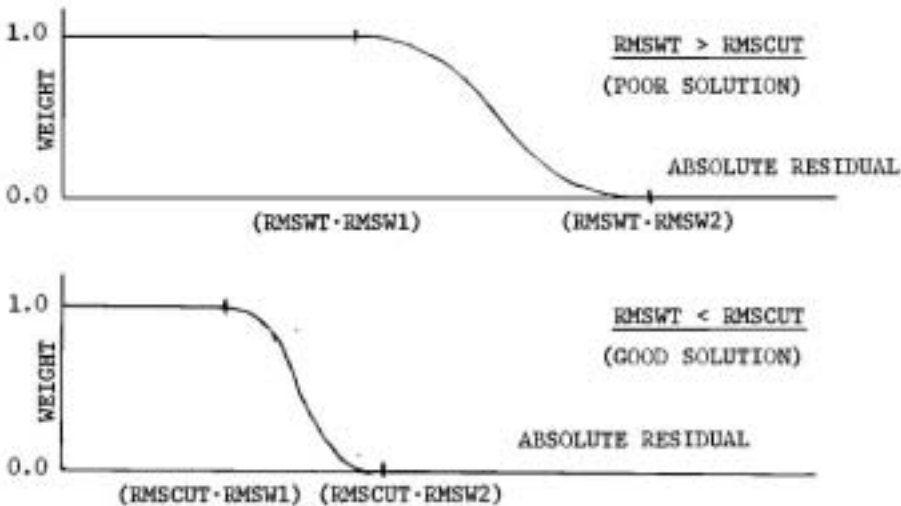
Figure 7. The residual weighting function.

RMSCUT, RMSW1 and RMSW2 are constants set with the RMS command. RMSWT is the root-mean-square travel time residual computed before residual weighting is applied. If RMSWT is larger than RMSCUT (upper figure), the function stretches out and scales with RMSWT such that most of the network stations receive weights somewhere in the tapering part of the function. If RMSWT is smaller than RMSCUT (lower figure), the function is fixed. Stations with residuals larger than RMSCUT x RMSW2 then receive no weight.

SOME SIMPLE COMMAND SEQUENCES

Several examples of command sequences will illustrate the flexibility of Hypoinverse. The intent here is to point out some of the most useful commands and how they might be sequenced.

Example 1.   The simplest possible run (keeps all other defaults).

```
CRH 1 'MOD1.CRH'        Read layer model 1 from the file MOD1.CRH.
STA 'ALL.STA'           Read station list from file ALL.STA.
PRT 'RUN1.PRT'          Send printer output to the file RUN1.PRT.
PHS 'SET1.PHS'          Define the phase file as SET1.PHS.
FIL                     Determine the phase file format from the file.
LOC                     Locate the events.
STO                     Stop the program.
```

Example 2. Generates additional output files.

```
CRT 1 'MOD1.CRT'        Read gradient model 1 from the file MOD1.CRT.
STA 'ALL.STA'           Read station list from file ALL.STA.
PRT 'RUN1.PRT'          Send printer output to the file RUN1.PRT.
SUM 'SET1.SUM'          Write Hypoinverse summary data to the file SET1.SUM.
ARC 'ARC1.ARC'          Write archive data to the file ARC1.ARC.
PHS 'SET1.PHS'          Define the phase file as SET1.PHS.
LOC                     Locate the events.
```

*Example 3. Read archive format phase data, write a compact print file, and use HYPO71 format
          summary output, all with Y2000 formats.*

```
200 T 1900 0            Enables Y2000 formats, sets some defaults
CRH 1 'MOD1.CRH'        Read layer model 1 from the file MOD1.CRH.
H71 2 2 2               Use HYPO71 summary, terminator & station formats.
STA 'STAS.HYPO71'       Read station list.
PRT 'RUN1.PRT'          Send printer output to the file RUN1.PRT.
LST 0                   Don't list available stations or crust model in printout.
KPR 1                   List only final solution & station data on printout for each event.
TOP F                   Doesn't begin each new event at the top of a page.
SUM 'SET1.SUM'          Write summary data to SET1.SUM.
COP 3                   Read archive format input.
PHS 'OLD.ARC'           Define the phase file as OLD.ARC.
LOC                     Locate the events.
```

Example 4. Locate a set of events with, then without S.

```
CRH 1 'MOD1.CRH'        Read layers model 1 from the file MOD1.CRH.
STA 'ALL.STA'           Read station list from file ALL.STA.
PRT 'RUN1.PRT'          Send printer output to the file RUN1.PRT.
SUM 'WITHS.SUM'         Put summary data with S in this file.
SWT 1.0                 Set S weighting to 1.0 (full weight).
PHS 'SET1.PHS'          Define the phase file as SET1.PHS.
LOC                     Locate the events.
SUM 'NOS.SUM'           Put summary data without S in this file.
SWT 0                   Set S weighting to 0 (no weight).
PRT 'RUN2.PRT'          Send printer output for the second run to this file.
LOC                     Locate the same events as before.
```

COMMANDS RECOGNIZED BY HYPOINVERSE

The commands are grouped by function, and thus all the commands dealing with amount of printout are listed together, for example. The required parameters are listed below each command. The function of many commands is discussed above in the sections on various inputs and calculations. The parameter defaults, if any, are listed in the examples. If you do not supply parameters on the command line, you will be prompted for parameters and shown the current value. If you want to keep the value, just press the RETURN key. Type HELP or "?" for a listing of commands and a very brief description, or type MORE for additional commands.

All of the commands are indexed. The index thus contains an alphabetical command list, with bold face page numbers referring to this command dictionary.

Unless indicated otherwise by "RIGHT NOW", commands set names or parameters without reading in data or doing any processing. Normally one issues several commands, some of which read files "RIGHT NOW", then ends with the LOC command to start processing.

INPUT FILES

CRT   Read a linear gradient crustal model expressed as a travel-time table RIGHT NOW. See the section on crustal models. The table is generated by the program TTGEN from a series of velocity-depth points.

- Model number (1-36).
- File name containing the travel-time table.

Example: `CRT 2 'REGION2.CRT'` [there is no default]

CRH   Read a homogeneous layer crustal model RIGHT NOW. See the section on crustal models.

- Model number (1-36).
- File name containing the layer depths and velocities.

Example: `CRH 3 'LAYMOD3.CRH'` [there is no default]

STA   Read a station data file into memory RIGHT NOW. See the H71 command for station file formats.

- Supply file name containing stations.

Example: `STA '1983.STA'` [there is no default]

PHS   Set the phase data input filename. See the COP command for selecting phase format. In the special case of reading many CUSP MEM files with the COP 7 command, supply the file listing the CUSP ID numbers to locate.

- Phase filename.

Example: `PHS 'PHASES.PHS'` [there is no default]

BINARY FILES

Binary files are designed for fast reading of station and crust data. You would read in all the ASCII files once, write them out in binary, then read data in binary each time you use the program. History data that changes with time, such as calibration factors or station magnitude corrections, are only read from the ASCII files.

WCR   Write a snapshot of all crustal models and multiple model definitions currently in memory to a file in binary form, RIGHT NOW. You should have issued all CRT, CRH, NOD, ALT and MUL commands first.

- Supply the filename to write to.

Example: `WCR 'home/calnet/klein/hypfiles/multmod.bin'`

RCR   Read a binary file of all crustal models and multiple model definitions previously written with a WCR command, RIGHT NOW. RCR replaces the CRT, CRH, NOD, ALT and MUL commands. Binary reads are several times faster than ASCII reads and worth the effort for frequently read files.

- Supply the filename to read from.

Example: `RCR 'WE:[ KLEIN.MULT] MULTMOD.BIN'`

WST   Write a snapshot of all station data and delays currently in memory to a binary file RIGHT NOW. You should have issued all STA and DEL commands first. Note that only one calibration factor and one type of each magnitude correction are written (no histories or expiration dates) so the ATE, CAL, FMC and XMC commands must still be used.

- Supply the filename to write to.

Example: `WST '/home/klein/hypfiles/allcrust.bin'`

RST   Read a binary file of all station data and delays previously written with a WST command RIGHT NOW. RST replaces the STA and DEL commands. Binary reads are several times faster than ASCII reads and worth the effort for frequently read files.

- Supply the filename to read from.

Example: `WST '/home/klein/hypfiles/allcrust.bin'`

READING ADDITIONAL STATION DATA

The following 5 commands can't be given until a station file is read with the STA command. That is because these commands read station data and store it only for the stations already in memory. All data read by these commands may also be supplied in the station file, but some fields will be redefined by these commands. Data management is much easier with station locations (STA command), delays (DEL), attenuation (gain) settings (ATE or CAL), duration magnitude corrections (FMC) and amplitude magnitude corrections (XMC) in separate files. For example, the network delays may all be stored in one place and need not be incorporated into every station file. In addition, one file can have delays for every conceivable station but only those matching stations in the location file will use memory in Hypoinverse.

DEL   Read in the station delays in seconds for either one or all models from a file, RIGHT NOW. Also read in the stations to use with alternate crustal models. The delay file, if used, must be read in after the station file because the station list must already be in memory. The stations in the delay file need not correspond in number or order to those in the station file. (Note that the old DLY command no longer functions because DEL is a more flexible way of assigning delays.) Also see the "multiple crustal models" sections for the meaning of multiple delays. See the section on station formats for the format of this file.

     - Specify a code selecting which type of delay file to read:

- 1-36, model number to which delays apply (specify one delay per station).
- 0, read in the alternate status codes and delays for all models (all models on same line).
- -1 (minus one), read in the alternate crust model code that states which stations use an alternate model.

     - Specify the file name with the station delays.

     Example: `DEL 0 'ALL.DEL'` or `DEL 1 'MOD1.DEL'` (There is no default).

ATN   Say whether the "CAL FACTOR" data field on the station record (read with the STA command) contains an attenuation or cal factor (see amplitude magnitude section for explanation of these terms):

- T The field is an attenuation setting.
- F The field is a calibration factor.

     Example: `ATN F`

ATE   Read station attenuations (related to gain) from a history file, RIGHT NOW. You must read your station file with the STA command before reading the station attenuation file. Once an attenuation file has been read and a set of attenuations defined for a certain date, the file is re-read as needed during a location run to find a new attenuation value after the

expiration date of the previous attenuation value. The new value is used until it expires. This is why the earthquakes must be in chronological order when this feature is used. See the section on specifying the station list for the file format. Also see the "event magnitudes and names" section.

- Supply the filename of the attenuation history file.
- Supply the date and time (year, month, day, hour) for which to load the initial station attenuations. Hypoinverse will update the station attenuations as needed. Use a year of zero to load the earliest attenuations.

Example (there is no default): ATE 'ALL.ATN' 1980 1 1 0

CAL   Read station calibration factors (gains) from a history file, RIGHT NOW. You must read your station file with the STA command before reading the station calibration factor file. Once a calibration factor file has been read for a certain date, the file is re-read as needed during a location run to find a new calibration factor after the expiration date of the previous value. The new value is used until *it* expires. This is why the earthquakes must be in chronological order when this feature is used. See the section on specifying the station list for the file format. Also see the "event magnitudes and names" section.

- Supply the filename of the calibration factor history file.
- Supply the date and time (year, month, day, hour) for which to load the initial station calibration factor. Use a year of zero to load the earliest calibration factor, and let Hypoinverse update the station calibration factor from the file as needed.

Example (there is no default): CAL 'ALL.CAL' 1980 1 1 0

FMC   Read station duration magnitude corrections from an FMC history file, RIGHT NOW. You must read your station file with the STA command before reading the magnitude correction file. Once a file has been read, it is re-read as needed during a location run to find a new magnitude correction after the expiration date of the previous one. See the section on specifying the station list for the format. You can suppress the coda magnitude for a given station from being used in the event magnitude by adding 5.0 to the magnitude correction. You can suppress the coda gain correction for a given station from being used by adding 10.0 to the magnitude correction.

- Supply the filename of the duration magnitude correction history file.
- Set the flag governing where the station fmag weights come from:
    ▪ T to use the fmag weights on the station line (the default).
    ▪ F to give an fmag weight of 1.0 to all stations present in the correction file, and 0.0 to all the stations that are not.
- Supply the date and time (year, month, day, hour) for which to load the initial station magnitude corrections. Use a year of zero to load the earliest corrections and let Hypoinverse update the magnitude corrections as needed.

Example (there is no default): FMC 'ALL.FMC' T 80 1 1 0

XMC Read station instrument types and amplitude magnitude corrections from an XMC history file, RIGHT NOW. You must read your station file with the STA command before reading the magnitude correction file. Once a file has been read, it is re-read as needed during a location run to find a new magnitude correction after the expiration date of the previous one. See the section on specifying the station list for the format. You can suppress the amplitude magnitude for a given station from being used in the event magnitude by adding 5.0 to the magnitude correction.

    - Supply the filename of the amplitude magnitude correction file.
    - Set the flag governing where the station xmag weights come from:
        ▪ T to use the xmag weights on the station line (the default).
        ▪ F to give an xmag weight of 1.0 to all stations present in the correction file, and 0.0 to all the stations that are not.
    - Supply the date and time (year, month, day, hour) for which to load the initial station magnitude corrections. Use a year of zero to load the earliest corrections and let Hypoinverse update the magnitude corrections as needed.

    Example (there is no default): `XMC 'ALL.XMC' T 80 1 1 0`

FILE FORMATS AND RELATED CONTROLS

200 Choose whether to use Y2000 formats. If you use Y2000 formats, you must do it for all input and output file types. The only files which have different Y2000 formats are those with dates: stations files, for example, are the same whether in Y2000 format mode or not. The exception is the traditional USGS phase format, which is always in its old format even when the program is in Y2000 format mode. Thus, if you have selected Y2000 formats with the "200" command and are reading traditional phase format, a default century (1900 or 2000) is added to all the 2-digit years, and full 4-digit years are used for all output files. I have written a separate program called 2000CONV to convert old Hypoinverse files to their new formats.

    - Supply the format flag: T for Y2000 formats, F for old formats.
    - Supply the default century to use when reading the traditional USGS phase format.
    - Supply the default amplitude units code when reading the traditional USGS phase format. The Y2000 archive format has a field (that is not available in the old phase format) describing the units (mm, counts, etc.) of amplitude measurement. 0 (zero) is the default code for the traditional mm units.

    Example: `200  F 1900 0`

COP Set the input phase data format. See the phase data input format section for the formats. Shadow records contain additional data, which is not processed by Hypoinverse, but just passed through. Shadow records begin with a "$" character and follow each header, phase or terminator line.

- Supply the format number as follows:

1) Traditional USGS (full) phase format. (No Y2000 format, 2-digit years only).
2) Unused
3) Hypoinverse archive format.
4) Traditional phase format with shadow records (PHASEOUT HYPO71 shadow option).
5) Archive format with shadow records.
6) One CUSP event (supply CUSP ID number with LOC command).
7) Many CUSP events (ID numbers read from a file; see FID command).

For options 6 and 7: See the phase input section for discussion of CUSP input and MEM files. CUSP input applies to VAX version only. Also specify the level of output to the MEM file being read (for example, `COP 6 0`):

- ▪ 0 No MEM file output.
- ▪ 1 MEM output to data structures.
- ▪ 2 MEM output to shared memory region while timing events.
- ▪ 3 MEM output to disk file.

Example: `COP 1` (the default)

FIL   Determine the format of the input phase file already set with the PHS command automatically, RIGHT NOW. The first 2 lines of the phase file are read when you issue the FIL command, and the input and output format numbers and Y2000 format mode (normally set with the 200, COP and CAR commands) are reset. Does not work for binary MEM files.

As a convenience, FIL also determines the format of the various types of summary files (Hypo71-2000, etc.), which are not valid Hypoinverse input files.   The results are announced without resetting the input format number.

Example: `FIL` (there are no parameters)

CAR   Set the archive file output format. The format types correspond to some of the phase data input formats.

- Supply the format number as follows:

1) Archive format, no shadow lines.
2) Unused.
3) Archive format, write a shadow line after every line.

Example: `CAR 1` (the default).

FID   Set the fortran format for reading CUSP ID numbers to locate (use with COP 7). This is determined by the format of the file identified with the PHS command, for example a MEM file list.

- Supply the format as a text string.

Examples: FID '(I10)' (the default) or FID '(1X,I8)' (for current NCSN files)

H71   Choose between HYPOINVERSE and HYPO71 formats where they differ.

- Set the summary output format: 1 for HYPOINVERSE, 2 for HYPO71.
- Set the terminator input format:
1) Hypoinverse
2) HYPO71
3) Get the trial hypocenter from the Hypoinverse header record (COP formats 3 or 5 only).

- Set the station input format:
1) Old Hypoinverse format #1
2) HYPO71
3) New Hypoinverse format #2

Example: H71 1 1 1 (the default)

LET   Choose how many letters to match when finding correspondences between the station codes given in the various files such as station location, phase, and the various correction files. Supply the number of letters for each of the site (max 5), network (max 2) and component (max 3) codes. A value of 0 means no characters are tested, and any characters will match. Normally you would specify the maximum number of letters your network actually uses.

In what follows, "-" is just a visible separator. Thus for a code like (CALB -NC-VHZ) you would use LET 5 2 3. If you used LET 5 0 3, you would also match codes like (CALB - -VHZ) or (CALB -**-VHZ). The full code from the station location file is the one used on output files. Thus if (CALB -NC-VHZ) were in the station file and (CALB - -VHZ) were in the phase file and LET 5 0 3 were used, the station would match and (CALB -NC-VHZ) would appear in the archive and print files. Note that station delays (from the DEL files) are always matched to all components of the station site and network as if 0 were used for the comparison length of components.

- Supply the number of station site code letters to test in making matches (2-5).
- Supply the number of station net code letters to test in making matches (0-2).
- Supply the number of station component code letters to test in making matches (0-3).

Example: `LET 4 0 0`

UNK Set a list of 4-letter station codes (or the first 4 of 5 letters) that you expect to be in the phase but not station file. Stations in this list will not produce an "unknown station" error message but can be archived in the ARC file (see KEP command). Other unknown station codes not on the "UNK" list can be output to the archive file, but will see an error message each time they are encountered.

- Specify the number of stations (maximum 10), and that many 4-letter codes. Stations will be recognized as "unknown" by a match of the first 4-letters of their 5-letter code, but all 5 letters will be used when the data is written. This is very useful for traces such as time codes where the data is to be archived but no true station or measured phase exists.

Example: `UNK 0` (the default) or `UNK 3 'ABCM' 'IRIG' 'WWVB'`

KEP Say whether to keep phase information from unrecognized stations and output them to the archive file. Stations mentioned in the UNK command are also under control of the KEP command and can be saved or not.

-Specify T to write unknown stations to the archive file, or F to eliminate them.

Example: `KEP T`

LAB Each event may have a 1-letter label code that marks its processing status, or labels the run to distinguish it from other runs. This is currently in col. 153 (col. 163 of the Y2000 version) of the Hypoinverse summary format. This label can either be passed through from the input archive file, or a new label assigned to all events in this location run.

- Supply a 1-letter run label code (blank is the default).
- Set a flag T to pass the old 1-letter run label code from archive input to output files, or F to insert the new label code into the output files.

Examples: `LAB ' ' T` (the default) or `LAB 'A' F`.

OUTPUT FILES

PRT Set the print output filename.

- Supply the filename. Use 'NONE' or 'none' to omit a printout file.

Example: `PRT 'NONE'` (the default), or `PRT 'PRTFIL.PRT'`

SUM Set the Hypoinverse summary output filename.

- Supply the filename. Use 'NONE' or 'none' to omit a Hypoinverse summary file.

Example: SUM 'NONE' (the default), or SUM 'OUT.SUM'

ARC Set the archive output filename. This file contains all the data calculated for each station, and can be read back in for a later relocation.

- Supply the filename. Use 'NONE' or 'none' to omit an archive file.

Example: ARC 'NONE' (the default), or ARC 'OUT.ARC'

MFL Set the magnitude data output filename. This file contains precise magnitudes and other data necessary to recalculate magnitudes or evaluate magnitude statistics.

- Supply the filename. Use 'NONE' or 'none' to omit a magnitude file.

Example: MFL 'NONE' (the default), or MFL 'OUT.MFL'

ERF Error messages (for bad data, station names, etc.) are always written to the print file if one is specified. They may also be sent to the terminal to spot errors during a location run by turning error reporting on. The most serious errors are always sent to the terminal.

- Supply a T to send error messages to the terminal, F otherwise.

Example: ERF F (the default).

APP Set the 3 logical flags that indicate whether an existing output file is appended to (T), or whether a new one is created (F).

- Supply 3 logical flags for: 1= printout file, 2= summary file, 3= archive file.

Example: APP F F F (the default), or APP F T F to append only to the summary file.

MULTIPLE CRUSTAL MODELS

MUL Indicate whether region-dependent crustal models are to be used. If true, also give the number of the default model to use outside the explicit regions.

- Set a flag T to use multiple models or F to use one model.

Example: MUL F (the default) or MUL T 1

NOD Define a circle on a map (by its center and radius) within which all epicenters will use a particular crustal model and set of station delays. Presently 124 nodes or circles are allowed. Each NOD command issued defines a new node: you thus cannot reset or examine a node once defined. The SNO command displays the current nodes on the

terminal. The transition width defines an annulus outside the inner circle within which the crustal model is used in combination with other models.

- Set the latitude of the circle center (degrees, positive north).
- Set the latitude of the circle center (minutes).
- Set the longitude of the circle center (degrees, positive west).
- Set the longitude of the circle center (minutes).
- Set the radius (km) of a circle where the model is used exclusively.
- Set the transition width outside the circle (km) for partial weighting.
- Set the crust model number for this node.

Example: (there are no defaults) `NOD 37 10 122 5.4 30 10 2`

(This node for model 2 is a 30 km radius circle with a 10 km-wide transition zone surrounding it).

SNO   Show the nodes (centers and radii of circular regions) for various crustal models, which have been defined by NOD commands. There are no parameters.

Example: `SNO`

ALT   Designate one crustal model as an alternate to another, and use the alternate model for stations so indicated in the station file. If a station is not specially marked, it will use the primary models; if it is marked as "alternate", it will use the alternate model if one is defined. Alternate models allow using two different models with two different sets of stations for the same earthquake, and may be used with or without multiple models for different regions. Any number of models may have alternates. An alternate model must be of the same type (layer or gradient) as the primary model.

- Specify the primary model number to have an alternate.
- Specify its alternate model number.

Example: (there are no defaults) `ALT 2 3` and `ALT 4 5`

In this example, an epicenter falling in the region (defined by NOD commands) for model 2 will use model 2 for normal stations, and model 3 for stations marked as alternates. An epicenter in region 4 will use model 4 for the normal stations and model 5 for the <u>same</u> set of alternate stations. For example, NCSN uses this feature for two model regions that straddle the San Andreas Fault. The stations to the east of the fault are designated as alternate. These alternate stations use the alternate models from every region where an alternate model is defined, namely 3 and 5. Because the fault is a velocity discontinuity, two different models are used for earthquakes on the fault: the "normal" model for stations and rays to the west, and the "alternate" model for stations and rays to the east.

PROCESS EVENTS IN A PHASE FILE

LOC   Locate events, RIGHT NOW. This is the command that actually locates earthquakes using the files and parameters set by previous commands. There are no parameters, except when locating individual CUSP events (COP 6), where you supply the CUSP id number as the parameter.

Example: `LOC`

BUG   Check phase file for format problems and station file for missing stations, and write error messages to the print file, RIGHT NOW. Phase, station, and print files must have been specified before issuing the BUG command. Works only with ASCII input formats. The BUG command has not been tested with all program options, so if you encounter a program "bug", try using the slower LOC command to debug your phase file.

Example: `BUG`

PRO   Interactively edit and relocate earthquakes in individual files, one event per file, RIGHT NOW. See the discussion above under "Interactive Earthquake Processing" and the BAS command below. There are no parameters.

Example: `PRO`

BAS   Set parameters needed for building the input and output filenames used in interactive processing started by the PRO command. The filenames consist (are a concatenation) of a base name unique to the event and a suffix (file extension) for each file type. The file extensions normally begin with a period. See the discussion above "Interactive Earthquake Processing".

- Supply the filename listing events to be processed. For each event, the file must contain the base name (text string) used to form the I/O filenames for the event. The base file name is read from this file with the format specified with the FID command. Normally base names are the dates (YMDHMS) of the earthquakes, and the file is a summary file.
- Number of characters in the base name. Base names are fixed in length.
- Format for reading base names from the file named above. This is an ASCII string, and must contain parentheses as required by fortran.

The following four parameters are the file suffixes for the file types read or written by Hypoinverse for each event. The archive and summary file types are optional and may be suppressed by using "NONE" or "none" as a suffix string.

- Input (phase or archive) file extension.
- Archive output file extension.
- Summary output file extension.
- Print output file extension.

Example: `BAS 'LISTFIL.' 12 '(A12)' '.PHS' '.ARC' '.SUM' '.PRT'`

(the default) or

```
        BAS 'LISTFIL.8401' 12 '(A12)' 2*'.ARC' 'NONE' '.PRT'
```

PRINTED OUTPUT FORMAT

LST   List stations, crust and test parameters at the beginning of the print output file.
- Set the print code:
  - 0 Print earthquakes only.
  - 1 Add the location parameters & filenames to beginning of printout.
  - 2 Add a station list and all crust models.

If the print code is 2, add two more parameters:
- Set the station detail code:
  - 0 List no stations.
  - 1 List station locations, cal factors, first delay etc (1 line per sta)
  - 2 Also list delays for all crust models (adds 1 or 2 lines per station)

- Set the crust model detail code:
  - 0 List no crust models.
  - 1 List the layers, nodes and other data for each model.

Note that `LST 1` and `LST 2 0 0` produce the same result.

Example: `LST 1` (the default) or `LST 2 1 0`

KPR   Control the amount of information in the print file for each event.

- Supply KPRINT, which controls print output. Specifying a value also outputs all data output by lower values:

0) Print final location only (2 lines).
1) Add station list for final location.
2) Add the location & adjustments (one line) for each iteration.
3) Add the eigenvalues, covariance matrix & error ellipse. Also print a message each time an updated magnitude correction, attenuation or cal factor is loaded from one of the station history files.
4) Unused
5) Unused
6) Add the station list at each iteration.

Example: `KPR 3` (the default).

TOP   Start each earthquake at the top of a page in printout file.

- Set a logical flag (T or F) that controls whether to start each new event with a form-feed.

Example: `TOP T` (the default).

REP   Report each earthquake as it is located with a 1-line message on the terminal, and control whether un-weighted stations are listed in the print file.

- Set a logical flag (T or F) that controls whether to report basic data of each event as located to the terminal. Data includes time, location, preferred magnitude and id number.
- Set a logical flag (T or F) to control the printing of un-weighted stations to the print file. If the flag is T, a station with zero-weighted P, S, coda or amplitude will not print but will be written to the archive file.

Example: `REP T F` (the default).

TRIAL DEPTH, VELOCITY RATIO & ERRORS

ZTR   Set the trial depth in km for the run, which can be over-ridden for individual events on their terminator lines. If the trial depth is negative, earthquakes are held fixed at this depth.

- Supply the trial depth ZTR in km.

Example: `ZTR 7` (the default).

POS   Set the P to S velocity ratio POS. All S travel times are calculated as POS times the model travel time (P velocities assumed). S station delays are derived from P delays by multiplying by POS.

- Supply the P to S velocity ratio POS.

Example: `POS 1.75` (the default).

ERR   Set the assumed reading and timing error in seconds. This should be the total error from all sources including the reading error and all un-modeled crust and delay time errors. A good value to use is your typical RMS residual after you have adequate crust and station delays. See the ERC command.

- Supply the reading and timing error RDERR.

Example: `ERR .15` (the default).

ERC   Set the coefficient ERCOF of the RMS travel-time residual in the expression for the actual timing error: ACTUAL TIMING ERROR = $\sqrt{ (RDERR)^2 + (ERCOF*RMS)^2 }$ Set ERCOF according to the influence you want the fit of your data (the RMS) to play in the location error calculation. ERCOF should usually be in the range 0-1 inclusive. The calculated location errors will be proportional to this "actual timing error".

- Supply ERCOF, the error coefficient of RMS.

Example: `ERC 1.0` (the default).

MIN   Used to skip events with too few reporting stations. If fewer than MINSTA stations are present for an event, no location is even attempted. This is useful when small events are to be screened out. See also the JUN command.

- Supply MINSTA, the minimum number of stations required to attempt a location.

Example: `MIN 4` (the default).

CONVENIENCE AND CONTROL COMMANDS

INI   Initialize Hypoinverse by running a standard command file which the user has set up to set default variables, read in station and crust files, etc. The unix version of the program executes the command file named in the environment variable HYPINITFILE. Other versions have the file and pathname compiled into the program. The file is set in hybeg.f or hybeg.for. See the section "Initializing with your defaults and input files". Once set up, this makes starting the program easy.

Example: `INI`

HEL   Typing "?", HELP, HEL or HELL in Hypoinverse gets a brief listing of the most important commands. No detailed information is available through HELP.

MOR   Typing MORE or MOR gets a listing of additional commands that do not fit on the basic HELP screen.

@   Files of commands can be executed (as if they were typed at the keyboard) by typing @filename. A command file may call another command file (returning to where it left off) and be nested up to 4 levels deep.

Any operating system command can be executed from within Hypoinverse by typing #command. This has no effect on current parameters, and control returns to Hypoinverse when the command finishes. This can be used to edit files, check the directory for files, etc.

STO   Stop the program. There are no parameters.

SHO   List the current input and output files on your terminal.

MAX   List the current array maxima for stations, phases, crust models etc. This will show the maximum array capacity, not the number of stations that have actually been read. The

number of stations read is announced after the see the STA and related commands have finished.

INP Input phases data from the keyboard with station prompting. See the section on inputing phase data from the keyboard. There are no parameters.

TYP Type a line of text on the terminal. This can announce something when running a Hypoinverse command file, or give a hint about a command which will prompt for something. No string apostrophes are needed.

Example: `TYP Now reading stations...`

MAGNITUDES

MAG Select the type of coda magnitude relation and weighting to use for the two coda magnitudes calculated for each event.

- Set the magnitude type number MAGSEL for the primary coda mag FMAG1:
  1 Traditional duration (F-P) magnitude Md (Lee et al 1972; DUR command)
  2 Elapsed Time (tau) magnitude Mt (Michaelson 1987; TAU command)
  3 Second duration (F-P) magnitude (DUB command)

- Say whether to use the coda weight on the phase line (T), or to ignore any coda weight (F).
 o T Use the coda weight on the phase line. 0 or blank on the phase line is full weight, 1-3 are partial weights, and 4-9 is no weight.
 o F Ignore any coda weight. All codas are given full weight, except those with a weight code of "X" or "N".

- Set the magnitude type number MAGSE2 for the secondary coda mag FMAG2:
  1 Traditional duration (F-P) magnitude Md (Lee et al 1972; DUR command)
  2 Elapsed Time (tau) magnitude Mt (Michaelson 1987; TAU command)
  3 Second duration (F-P) magnitude (DUB command)

- Choose the log(Ao) relation (see amplitude magnitude section):
  1 Eaton (BSSA, 1992)
  2 Bakun and Joyner (BSSA, 1984)
  3 Richter's (1958) approximation
  4 UC Berkeley (Nordquist, BSSA, 1948)
  5 P amplitude (presently unused)

Example: `MAG 1 T 1 1` (the default)

PRE Establish the criteria for choosing the preferred magnitude for each earthquake from the five magnitudes potentially available. If a given magnitude is 0.0, it has not been computed and is never chosen as preferred. The criteria for choosing a magnitude

depend on its availability, number of stations used, and magnitude range. See the section on preferred magnitudes. The procedure is to run down your list of up to 10 choices until a magnitude is found that satisfies all criteria, then it is chosen as preferred. Note that a given magnitude type may appear more than once in your list if the criteria become more lax down your list. The preferred magnitude and its 1-letter type code are output to all files. The magnitude types (with their numbers) to choose from are:

1) Primary duration magnitude
2) Primary amplitude magnitude
3) An externally derived (sometimes called "Berkeley") magnitude
4) Secondary amplitude magnitude
5) Secondary duration magnitude
6) and 7) are reserved for un-implemented P-amplitude magnitudes

- Supply the number of possible magnitude types (which follow) to evaluate as the preferred magnitude.
- For each of the magnitudes to evaluate, supply:

- Magnitude type (1-5).
- Minimum number of readings to qualify.
- Minimum magnitude to qualify.
- Maximum magnitude to qualify.

Example: `PRE  0` (no preferred magnitude, the default), or (in this example I use commas to visually separate each set of magnitude criteria, but any separator would do)

`PRE 3, 4 4 4 9, 3 0 0 9, 4 0 0 9` (magnitude type 4 if 4 or more readings and M>4, then any magnitude of type 3, then any magnitude of type 4)

CODA DURATION MAGNITUDES

FC1   Define the station components to include in the primary coda magnitude FMAG1 for an event. A maximum of 10 different components may be used. The components will select which of the station magnitudes are used for the earthquake magnitude. Also set the 1-letter label for FMAG1.

- Supply the 1-letter label code for FMAG1.
- Supply NCPF1, the number of components to use in calculating FMAG1:
Set NCPF1 = -1 to use all components in FMAG1.
Set NCPF1 = 0 to use no stations in FMAG1.
Set NCPF1 = the number of components to use for FMAG1 (1-10).
- Supply the NCPF1 3-letter component codes to use in FMAG1.

Examples: `FC1 ' ' −1` (use all components used for FMAG1, which has no label, the default) or

`FC1 'D' 1 'VHZ'` (use VHZ component stations for the "D" magnitude)

FC2   Define the station components to include in the secondary coda magnitude FMAG2 for an event. A maximum of 10 different components may be used. The components select which stations are used for the earthquake magnitude. Also set the 1-letter label for FMAG2.

- Supply the 1-letter label code for FMAG2.
- Supply NCPF2, the number of components to use in calculating FMAG2:
Set NCPF2 = -1 to use all components in FMAG2.
Set NCPF2 = 0 to use no stations in FMAG2.
Set NCPF2 = the number of components to use for FMAG2 (1-10).
- Supply the NCPF2 3-letter component codes to use in FMAG2.

Examples: `FC2 ' ' 0` (do not calculate FMAG2; 0 components) or

`FC2 'Z' 2 'VHZ' 'VLZ'` (define 2 components for "Z" magnitude)

FCM   Define duration magnitude corrections for named station components. These corrections are in addition to all other corrections. If you find that low gain horizontal stations typically give too large a magnitude, for example, you can apply a correction to all stations of that type without having to specify a correction for each station individually.

- Supply the number of components that follow with individual duration magnitude corrections (0-4).
- Supply pairs of 3-letter component codes and duration magnitude corrections.

Examples: `FCM 0` (no component corrections, the default), or

`FCM 2 'VLE' -0.3 'VLN' -0.3` (negative corrections for two component codes)

DUR   Define the constants used to compute the first magnitude (FMAG1) from coda duration (F-P) time. Two formulae may be used spanning different ranges of coda duration. The formulae have the form:

Md = FMA + FMB*log(duration) + FMF*duration + FMZ*depth + FMD*distance + FMGN * (-log(CAL factor) + 0.6) + station corrections

- Supply FMA1, FMB1, FMZ1, FMD1, and FMF1 for durations less than FMBRK.
- Supply FMA2, FMB2, FMZ2, FMD2, and FMF2 for durations more than FMBRK.
- Supply FMBRK, the duration above which to use the second set of constants. Set FMBRK=9999 to only use the first set of constants.
- Supply FMGN (normally 0 or 1) to regulate the use of gain correction. Also see the DUG command to regulate whether gain corrections are used.

Example: `DUR -5.2 3.89 .013 .0037 0, -.9 2.026 .013 .0037 0,`
`210. 0` (for Hawaii, the default), or

`DUR -.87 2 0 .0035 0, 5*0, 9999 0` (traditional CALNET) or
`DUR -2.06 2.95 0 .001 0, 5*0, 9999 1` (CALNET low gain)
`DUR -.81 2.22 0 .0011 0, 5*0, 9999 1` (Eaton, 1992)

DUB   Define the constants used to compute the second magnitude (FMAG2) from coda duration (F-P) time. The parameters are exactly the same as the DUR command, but are used for the second duration magnitude. The parameters may have different values than the DUR command, or may be used with another set of stations (see the MAG and FC2 commands). No component corrections (FCM command) or additional distance corrections (DU2 command) are used with FMAG2.

DU2   Set additional constants for Eaton's special terms in the Md relation.

Recall Eaton's relation:

$$M_D(f\text{-}p) = -0.81 + 2.22*\log (f\text{-}p) + 0.0011*D + Stacor + G + Component\ correction$$

$$+ 0.005*(D - 40) \qquad \text{if } D < 40 \text{ km}$$

$$+ 0.0006*(D - 350) \qquad \text{if } D > 350 \text{ km}$$

$$+ 0.014*(Z - 10) \qquad \text{if } Z > 10 \text{ km}$$

- Supply the coefficient of the (D – DBRKM1) term
- Supply DBRKM1, the distance less than which to apply the (D – DBRKM1) term
- Supply the coefficient of the (D – DBRKM2) term
- Supply DBRKM2, the distance greater than which to apply the (D – DBRKM2) term
- Supply the coefficient of the (Z – ZBRKM) term
- Supply ZBRKM, the depth greater than which to apply the (Z – ZBRKM) term

Example: `DU2 6* 0` (apply no corrections, the default), or

`DU2 0.005 40, 0.0006 350, 0.014 10` (Eaton's Md formula)

DUG   Choose the components to which the duration magnitude gain correction is to be applied. To apply the correction, the parameter FMGN must also be set to 1 with the DUR command.

- Supply NDUG, the number of components to use with duration magnitude gain corrections:
= -1   apply to all components
= 0   apply to no components
= n   use n components (1-10 max).
- Follow with n 1-3 letter components.

Examples: `DUG -1` (the default) or

`DUG 4 'VHZ' 'VLE' 'VLN' 'VLZ'`

TAU Define the parameters for the lapse-time (tau, end-of-coda time minus origin time) magnitude scale Mt. This algorithm assumes that the duration (F-P time) is entered on the phase line and adds the calculated P travel time to it to get tau. The formula is:

$Mt = DMA0 + DMA1*log(tau) + DMA2*log\_(tau) + DMLIN*tau + DMZ*Z +$
$DMGN*G + station corrections$

where:
tau is the elapsed time (P travel time + coda duration)
Z is the (positive) depth
G is the gain correction = 0.05*atten - 0.0375 (atten = 12, 18 etc.)
Also G = -0.5*log(CAL factor) + 0.3

- Supply the 6 coefficients named DM in the above equation: DMA0, DMA1, DMA2, DMLIN, DMZ and DMGN.

Example: `TAU -1.03 2.10 0 0.00268 0 1`

AMPLITUDE MAGNITUDES

XCH Choose the way to select the stations to use for each of the two different amplitude magnitudes.

- Supply the flag to make the choice:
 T Use the component code to identify stations for amplitude magnitudes. Then use the XC1 and XC2 commands to choose the components.
 F Use the instrument type (from the station line or the amplitude magnitude correction file) to choose which of the two amplitude magnitudes to apply the station to for the event. Use the XTY command to choose the instrument types, but you will also use the XC1 and XC2 commands to set the label codes for the two amplitude magnitudes.

Example: `XCH T` (Use components to choose amplitude magnitudes)

XC1 Define the station components to include in the primary amplitude magnitude XMAG1 for an event. A maximum of 10 different components may be used. The components select which stations are used for the earthquake magnitude. Also set the 1-letter label for XMAG1. You would only be using this command (except for setting the magnitude label) if the XCH argument is T.

- Supply the 1-letter label code for XMAG1.

- Supply NCPX1, the number of components to use in calculating XMAG1:
Set NCPX1 = -1 to use all components in XMAG1.
Set NCPX1 = 0 to use no stations in XMAG1.
Set NCPX1 = the number of components to use for XMAG1 (1-10).
- Supply the NCPX1 1-letter component codes to use in XMAG1.

Examples: `XC1 ' ' -1` (all components used for XMAG1, the default) or

`XC1 'X' 4 'V ' 'E ' 'N ' 'Z '` (Use 4 components for "X" magnitude. Here only the first of the 3 letters is used.)

XC2   Define the station components to include in the secondary amplitude magnitude XMAG2 for an event. A maximum of 10 different components may be used. The components select which stations are used for the earthquake magnitude. Also set the 1-letter label for XMAG2. You would only be using this command (except for setting the magnitude label) if the XCH argument is T.

- Supply the 1-letter label code for XMAG2.
- Supply NCPX2, the number of components to use in calculating FMAG2:
Set NCPX2 = -1 to use all components in XMAG2.
Set NCPX2 = 0 to use no stations in XMAG2.
Set NCPX2 = the number of components to use for XMAG2 (1-7).
- Supply the NCPX2 1-letter component codes to use in XMAG2.

Examples: `XC2 ' ' 0` (do not calculate XMAG2; no components) or

`XC2 'L' 2 'WLE' 'WLN'` (use 2 components for "L" magnitude)

XTY   Set the seismometer instrument types to use with the two different amplitude magnitudes. Use the standard type numbers. You must supply all parameters to this command, even if you are not using some of the instrument types for a magnitude (i.e. there must always be 8 arguments to this command). You would only be using this command if the XCH argument is F. See the section on "seismometer instrument types" in the amplitude magnitude chapter for the type numbers available.

- Supply the number of instrument types to use with the first amplitude magnitude XMAG1 (0-3, or –1 for all types).
- Supply the first instrument type for XMAG1.
- Supply the second instrument type for XMAG1.
- Supply the third instrument type for XMAG1.

- Supply the number of instrument types to use with the second amplitude magnitude XMAG2 (0-3, or –1 for all types).
- Supply the first instrument type for XMAG2.
- Supply the second instrument type for XMAG2.
- Supply the third instrument type for XMAG2.

Examples: `XTY 0 3*0, 0 3*1` (do not use any instruments to choose amplitude magnitude, the default), or

`XTY 3 3 0 2, 2 0 2 0` (Use seismometer types 3, 0 and 2 for XMAG1, and types 0 and 2 for XMAG2. The comma is only used as a visual separator, and is not really needed.)

XCM   Define amplitude magnitude corrections for named station components. These corrections are in addition to all other corrections. If you find that low gain horizontal stations typically give too large a magnitude, for example, you can apply a correction to all stations of that type without having to specify a correction for each station individually.

- Supply the number of components that follow with individual amplitude magnitude corrections (0-4).
- Supply pairs of 3-letter component codes and amplitude magnitude corrections.

Examples: `XCM 0` (no component corrections, the default), or

`XCM 2 'VLE' -0.3 'VLN' -0.3` (negative corrections for two component codes)

LA0   (LA-zero) Choose the $\log(A_0)$ relation to use with specific components. The $\log(A_0)$ relation is the distance term in the amplitude magnitude relation. The MAG command chooses the default $\log(A_0)$ relation to use with all components not specifically mentioned here. The relationships are:

>   1   Eaton (BSSA, 1992)
>   2   Bakun and Joyner (BSSA, 1984)
>   3   Richter's (1958) approximation
>   4   UC Berkeley (Nordquist, BSSA, 1948)
>   5   P amplitude (presently unused)

- Supply the number of components that will follow with their $\log(A_0)$ relation.
- Supply pairs of 3-letter component codes, and $\log(A_0)$ relationship numbers.

Example: `LA0 0` (no special components, the default), or

`LA0 2 'WLN' 4, 'WLE' 4` (Use the Nordquist relation with 2 Wood-Andersons)

ATN   Select whether to assume that station records have CALibration factors (`ATN F`) or attenuation settings (`ATN T`). The attenuations (in db) are entered in place of CAL factors on the station lines and must be a multiple of 6. See the amplitude magnitude section for the conversion between CAL factors and attenuation values. If a value is entered that is not an integer multiple of 6, the CAL factor is left as zero, which means no

magnitudes will be computed. This must be entered before the STA command to know how to interpret the numbers when reading station lines.

- Set a flag F for Cal factors or T for attenuations.

Example: ATN F

MISCELLANEOUS COMMANDS

NET   Set the network number for assigning names to earthquakes based on their locations. This option requires that prior definition of earthquake regions be coded into the KLAS subroutine. If you do not have a net with defined names, use a NET of 0. Present nets are 1=Hawaii, 2=Northern California, 3=new Hawaii. See the QPLOT documentation for a list of regions. The 3-letter region codes are written to the summary, archive and print files. The full name is written to the print file.

- Supply the net number.

Example: NET 0 (the default).

WEIGHTING OF ARRIVAL TIMES

JUN   Specify whether to force a solution of small and poor (junk) events. If the number of stations remaining after distance and residual weighting are applied is less than the minimum number (set with the MIN command), the event will normally abort. Setting the junk flag to TRUE cancels all distance and residual weighting for events that would otherwise abort.

Example: JUN F (the default)

DIS   Set the parameters that govern progressive down-weighting of more distant stations. DMIN2 is the distance to the second closest seismic station in km. The distance weight is 1.0 for stations closer than DMIN2*DISW1, 0.0 for stations beyond DMIN2*DISW2, and is cosine tapered between those distances. If DMIN2 is smaller (closer) than DISCUT, DISCUT is used in place of DMIN2 in the above distances. This means there is an expanding circle of stations if the earthquake is outside the network, but the circle can't shrink too small for earthquakes under a dense part of the network. See the section on weighting and figure 6.

- Supply ITRDIS, the iteration on which distance weighting is to begin. Iteration continues at least until distance weighting begins.
- Supply DISCUT.
- Supply DISW1.
- Supply DISW2.

Example: `DIS 4 50 1 3` (the default) or

`DIS 4 15 3.5 7` (works well for NCSN).

RMS   Set the parameters that govern progressive down-weighting of stations with larger travel time residuals. RMS is the root-mean-square travel time residual in seconds. The residual weight is 1.0 for stations with residuals less than RMS*RMSW1, 0.0 for stations with residuals larger than RMS*RMSW2, and is cosine tapered between these residuals. If the RMS is smaller than RMSCUT, RMSCUT is used in place of the RMS in the above cutoff residuals. See the section on weighting and figure 7.

- Supply ITRRES, the iteration on which residual weighting is to begin. Iteration continues at least until residual weighting begins.
- Supply RMSCUT.
- Supply RMSW1.
- Supply RMSW2.

Example: `RMS 4 .16 1.5 3` (the default).

WET   Specify the weight factors for the P and S phase weight codes 0, 1, 2 and 3. Weight codes 4-9 always result in a weight factor of 0.0. Code 0 should get full weight (1.0).

- Specify the weight for code 0.
- Specify the weight for code 1.
- Specify the weight for code 2.
- Specify the weight for code 3.

Example: `WET 1.0 0.75 0.5 0.25` (this is the traditional scheme, the default), or

`WET 1.0 0.5 0.2 0.1` (this is a statistically better weighting function for the actual errors that typical seismogram readers use as they assign weight codes).

SWT   Set the S phase weighting factor. Use 1.0 to give all S readings full weight, and 0 to not to use any S readings.

- Supply SWT, the factor by which all S weights will be multiplied.

Example: `SWT 1.0` (the default).

ITERATION AND CONVERGENCE PARAMETERS

CON   Set parameters governing tests for convergence of iterations to a final earthquake solution. The solution is considered final as soon as either the hypocentral adjustment or RMS change falls below set limits, or the maximum number of iterations is exceeded. You may want to see the section on fixing hypocenters.

- Supply ITRLIM, the maximum number of iterations allowed. (Using a value of 0 is not the only requirement for fixing hypocenters.)
- Supply DQUIT. If the hypocentral adjustment falls below DQUIT km, iteration stops.
- Supply DRQT. If the change in RMS residual falls below DRQT seconds, iteration stops.

Example: `CON 20 0.04 0.001` (the default).

DAM   Set the iteration and damping parameters affecting hypocenter adjustments. See the section on iteration and damping.

- Supply DXFIX. Keep the depth fixed until the epicentral adjustment is less than DXFIX km.
- Supply DZMAX, the maximum depth adjustment in km allowed without a forced damping of the adjustment vector.
- Supply DZAIR. If the depth adjustment would place the hypocenter in the air, move the depth upward by this fraction toward the surface.
- Supply DAMP, the mandatory damping factor for all hypocenter adjustments.
- Supply EIGTOL, the minimum eigenvalue permitted before no hypocentral adjustment is taken in that eigenvalue's direction.
- Supply RBACK. See BACFAC.
- Supply BACFAC. If the RMS increases by more than RBACK from one iteration to the next, move the hypocenter by the fraction BACFAC back toward the last location and continue iterating.
- Supply DXMAX, the maximum distance adjustment in km.
- Supply D2FAR, the maximum distance for the second closest station in km before iteration is terminated. This prevents a distant earthquake from going out of control.

Example: `DAM 7 30 0.5 0.9 0.012 0.02 0.6 50 250` (the default)

OUTPUT FILE FORMATS

Print Output

The amount of information in the printed output can be varied somewhat. The LST command controls the quantity of station, crust model and parameter data at the beginning of the print output file before any earthquakes are located. The KPR command governs the level of data seen for each earthquake.

User or data errors are marked with a specific message that begins with "***". You can search for errors by searching for these 3 asterisks. Errors can also be reported to the terminal as locations proceed (see the ERF command).

The print file begins with a status line containing the date run and the run label set with the LAB command.

Station table

NAME	The 5-letter station site code.
NT	The 2-letter station network code.
COM	The 3-letter station component code.
C	The 1-letter station component code, normally treated as an abbreviation to the 3-letter component code
R	The 1-letter station remark from the station line (NCSN uses the first letter of the site code, which designates the region).
LAT	Degrees and minutes (north).
LON	Degrees and minutes (west).
PDLY1	P delay for the first model. Model code appears above the column.
A	An "A" here designates stations for use with alternate models.
FCOR	Duration magnitude correction.
FWT	Duration magnitude weight factor for the station.
FMC.EXPIRE	Expiration date of the duration magnitude correction
XCOR	Amplitude magnitude correction.
XWT	Amplitude magnitude weight factor.
PSWT	Station weight factor for P and S phase times.
CAL	Calibration factor (related to gain).
CAL.EXPIR	Expiration date (Y,M,D,H) of CAL factor. 0 means no expiration date.
PER	Default period at which peak amplitudes are measured.
TYP	Station seismometer instrument or response type code.
PDLY2	P delay for the 2nd model. Model code appears above the column.

Part two of the station table (if requested) lists the 4-letter station codes, components, and delays for all of the other crustal models.

Next, the data listed for each crustal model includes the velocity at the top and depth to the top of each layer. If regional crustal models are in use, a listing of all geographic nodes assigned to the model is given. Each node is indicated by its number, center latitude and longitude, and the inner and outer radii of the transition zone surrounding the node. The transition zone is the annulus in which the weight of the model (relative to the surrounding models) is decreasing from 1.0.

Next, several of the most important parameters set by the various commands are listed. This includes the names if the input and output files.

Earthquake iteration output

The print output for each earthquake begins with a line with the event's date, sequence number (numbered sequentially within this run, but not output in the archive or summary file) and id number (it is read and written, and serves as a permanent and unique id number).

In the earthquake output, one line of information per iteration is printed when the print control set with the KPR command is 2 or larger. The final location data is always written to a print file, and the station list data is written if the print control is 1 or larger.

I	Iteration number.

ORIGIN	Seconds part of origin time.
LAT N	Latitude.
LON W	Longitude.
Z	Depth.
NWR	Number of P & S readings with final station weights larger than 0.1. This reflects all weight factors, including the distance and residual weights, which can vary from iteration to iteration.
RMS	The root-mean-square travel time residual using station weights.
DT	Origin time adjustment, applied to ORIGIN to get to the next iteration location.
DLAT	Latitude adjustment in km, positive north.
DLON	Longitude adjustment in km, positive west.
DZ	Depth adjustment, positive down.
RR	Length of adjustment vector in km.
NF	Number of free hypocenter parameters solved for and adjusted. Normally 4, but will be 3 on the first iteration since depth is held fixed. If N is less than 4, it is because the solution is poorly constrained and one or more eigenvalues are less than EIGTOL (see the DAM command).
MOD	The primary (largest weight) crustal model code for this epicenter.

Next you will see the error ellipse data. For each of the three principal errors that form the major axes of the error ellipse, SERR is the one-standard-deviation error in km, AZ is its azimuth in degrees east of north and DIP is its dip in degrees.

Final location data
The final hypocenter data is below a printed row of dashes.

YEAR MO DA	Date.
ORIGIN	Hour, minute, second.
LAT N	Degrees and minutes.
LON W	Degrees and minutes.
DEPTH	In km.
RMS	The root-mean-square travel time residual that uses all weights.
ERH	The horizontal location error, defined as the length of the largest projection of the three principal errors on a horizontal plane. The principal errors are the major axes of the error ellipsoid, and are mutually perpendicular. ERH thus approximates the major axis of the epicenter's error ellipse.
ERZ	The depth error, defined as the largest projection of the three principal errors on a vertical line.
XMAG	Primary weighted median amplitude magnitude (XMAG1).
FMAG	Primary weighted median coda magnitude (FMAG1).
PMAG	The preferred magnitude chosen from the available magnitudes by the PRE command.
NSTA	The number of stations (phase lines) read for this event.
NPHS	The number of phases (P and S) read for this event.
DMIN	Distance to the nearest station.

MODEL   The dominant crustal model code for the event. If some stations use an alternate model, a "*" follows the code.

GAP   The largest azimuthal gap between azimuthally adjacent stations.

ITR   Number of iterations required for the solution.

NFM   Number of P first motions reported for this event.

NWR   Final number of P & S readings with weights larger than 0.1.

NWS   Number of S readings with weights larger than 0.1.

NVR   Number of valid P & S readings (assigned weights larger than 0).

REMARKS   The first remark is the 3-letter region code based on location and depth.
               Two auxiliary 1-letter event remarks follow, and they are blank if there is nothing special about the event. The first is assigned by the analyst and may include F (felt), Q or B (quarry blast), R or N (explosion), T (tremor associated) or L (long period).
               Hypoinverse assigns the second remark. A pound sign (#) indicates convergence problems with the final solution such as running out of iterations, or failure of the hypocenter to reach a minimum in RMS. A minus sign (-) indicates the depth was held fixed either by the user or by insufficient data. An (X) indicates the location was fixed to the trial hypocenter.

AVH   The three status remark codes for the event found on the input archive header record (and passed through to output).
               o   A is the "authority" code, i.e. what network furnished the information. Hypoinverse passes this code through.
               o   V is the "version" of the information, i.e. what stage of processing or completeness the event is in. This can either be passed through, or can be set by Hypoinverse to all events in the current run to the code set with the LAB command.
               o   H is the "version number of last human review". Hypoinverse passes this code through. These 3 codes are in columns 114, 163 and 164 of the Y2000 summary line.

N.XMG   Number of primary amplitude magnitudes used in the event magnitude (XMAG1). It is the total of their weights, which is the same as the number if all weights are 1.

XMMAD   Weighted median-absolute-difference of the primary amplitude magnitudes. It is the error in the median magnitude, calculated this way because it is a median and not an average.

T   The label code given XMAG1 by the XC1 command.

N.FMG   Number of primary coda magnitudes used in the median (FMAG1). It is the total of their weights, which is the same as the number if all weights are 1.

FMMAD   Weighted median-absolute-difference of the primary coda magnitudes. It is the error in the median magnitude, calculated this way because it is a median.

T   The label code given FMAG1 by the FC1 command.

SOURCE   The code for the most commonly used data source:
              L   Location (P & S times).
              F   Duration magnitude.
              X   Amplitude magnitude.

The secondary magnitudes XMAG2 and FMAG2 and preferred magnitude PMAG will only
appear in the printout if they were calculated:

XMAG2   The secondary amplitude magnitude.
N.XMG2   Number of secondary amplitude magnitudes used in the median (XMAG2). It is
              the total of their weights, which is the same as the number if all weights are 1.
XMMAD   Weighted median-absolute-difference of the secondary amplitude magnitudes. It
              is the error in the median magnitude, calculated this way because it is a median.
T   The label code given XMAG2 by the XC2 command.
S   The principal data source code of XMAG2.

FMAG2   The secondary coda magnitude.
N.FMG2   Number of secondary duration magnitudes used in the median (XMAG2). It is
              the total of their weights, which is the same as the number if all weights are 1.
FMMAD   Weighted median-absolute-difference of the secondary duration magnitudes. It is
              the error in the median magnitude, calculated this way because it is a median.
T   The label code given FMAG2 by the FC2 command.
S   The principal data source code of FMAG2.

PREF.MAG   The preferred magnitude PMAG, chosen from the magnitudes available.
N.PMAG   Number of station magnitudes used in the median (PMAG). It is the total of their
              weights, which is the same as the number if all weights are 1.
PRMAD   Weighted median-absolute-difference of the magnitudes. It is the error in the
              median magnitude, calculated this way because it is a median and not an average.
T   The label code for the preferred magnitude.

If an external magnitude is present on the input header to the archive file, the next line gives
information about the external magnitude. It might be labeled "Berkeley mag".

REGION   The full name of the geographic region.
MODELS USED   If you are using multiple crustal models, the 3-letter codes of those
              actually used in the solution and their individual weights are given. There may be
              1, 2 or 3 models used depending on the node geometry at the epicenter.

Station list data

STA   Station 5-letter site code name. An asterisk after the station indicates that it uses
              the alternate crust model (see the ALT command and column 34 of the station
              format).
NET   The 2-letter network code.
COM   The 3-letter station component.
C   The 1-letter station component code, normally treated as an abbreviation to the 3-
              letter component code

R	The 1-letter station remark from the station line.
DIST	Epicentral distance.
AZM	Azimuth to station in degrees east of north.
AN	Angle if emergence at the hypocenter, in degrees up from nadir.
P/S	P or S remark code and P first motion.
WT	Assigned weight code and weight-out symbol (if any).
SEC	Observed arrival time.
TOBS	Observed travel time.
TCAL	Calculated travel time.
DLY	Station delay actually used, which might be a mixture of different crustal models.
RES	Travel time residual. The residual may be flagged (in the following column) by an "X" if that station had a time but no assigned weight, or an "*" if the residual is larger than 0.50 sec.
WT	Actual normalized weight used for this arrival, including assigned weight, distance weight, residual weight, station weight and global S weight. S phases are flagged with an "S" following the weight.
SR	The 1-letter data source code. The source usually refers to the contributing network or processing system.
R	The 1-letter seismogram remark carried from the phase data.
INFO	The information or importance contribution of this arrival to the solution. The total importance of all stations equals the number of unknowns (usually 4).
CAL	The calibration factor used for magnitude calculation.
DUR	Coda duration in seconds.
W	Duration magnitude weight code (0-4).
FMAG	Duration magnitude for this station.
T	The magnitude type code assigned this component by the FC1 or FC2 commands.
AMP	Peak-to-peak amplitude of the maximum part of the seismogram.
U	Amplitude units code derived from the units code on the input archive line: M millimeters, C digital counts, D HVO digital counts.
PER	Period (in sec) where amplitude was measured.
W	Amplitude magnitude weight code (0-9).
XMAG	Amplitude magnitude for this station.
T	The magnitude type code assigned this component by the XC1 or XC2 commands.

Magnitude data output

The magnitude data file contains some of the data in the archive file and more detailed data relevant to the magnitude calculations. The print output file only has some of the data used for calculating magnitude. All magnitudes in this file are written to two decimal places. Enough information is written, for example, to recalculate magnitudes or determine magnitude residuals as a function of time and station. Like the archive format, the data for an event begins with a

summary line and ends with a terminator line. Manipulation programs such as EXTRACT may thus be used with either the magnitude or archive file types. There is one line per station but only for stations reporting a non-zero duration or amplitude. The essential event data like date and event magnitude are written on every station line. The file may thus be sorted by station and each line has enough information to calculate a magnitude residual, for example. Magnitude corrections are also written to the file, so it should be possible to reconstruct the terms going into the final station magnitude. The additional oddball correction terms defined with the DU2 and XCM commands, however, are not written to the file. The format of the magnitude file is always Y2000 compatible.

Magnitude output format

Start Col.	Len.	Fortran Format	Data
1	5	A5	5-letter station site code. [STA]
6	2	A2	2-letter station network code. [SNET]
8	3	A3	3-letter station component code. [COMP3]
11	2	A2	2-letter station location code (component extension). [SLOCC]
13	1	1X	Presently blank.
14	4	I4	Origin time year. [KYEAR2]
18	8	4I2	Origin time month, day, hour and minute. [KMONTH…]
26	4	F4.2	Origin time seconds. [KQ]
30	4	F4.1	Epicentral distance in km. [KTEMP]
34	2	A2	P remark from phase line. [KPRK]
36	1	I1	P weight code. [LPWT]
37	4	F4.2	Calculated P travel time for this station. [MTCAL]
41	4	F4.2	P residual (observed TT = calculated TT + residual) [KPRES]
45	3	A3	3-letter event remark (epicentral location code). [REMK]
48	1	A1	Data source code for this station. [KSOU]
49	1	A1	1-letter station remark. [KRMK]
50	1	I1	Station type (0=Wood-Anderson, etc.) from station file. [JTYPE]
51	5	F5.3	Calibration factor from station or attenuation file. [JCAL]
56	3	3X	Presently blank.

Duration magnitude data for the STATION

59	4	I4	Coda duration (F-P time) for this station. [KFMP]
63	1	I1	Coda mag weight code for measurement (0-9, blank). [KFWT]
64	1	I1	Coda magnitude weight code for station (0-9, blank). [JFWT]
65	3	F3.2	Station duration magnitude correction. [JFCOR]
68	3	F3.2	Component correction for coda mag from FCM command. [ICOMF]
71	3	F3.2	Duration magnitude for the station. [KFMAG]
74	1	A1	Type label for duration magnitude (to tell which of the two possible relations was used). [LABF]
75	2	2X	Presently blank.

Amplitude magnitude data for the STATION

77	6	F6.2	Peak-to-peak amplitude. [AMPK]
83	1	A1	Units of amplitude (M=mm on Develocorder viewer or paper Wood-Anderson record, C=digital counts , D=digital counts (HVO only)). [CA1]

84	3	F3.2	Period at which amplitude was measured (sec). [KPER]
87	1	I1	Amp mag weight code for measurement (0-9, blank). [KXWT]
88	1	I1	Amp magnitude weight code for station (0-9, blank). [JXWT]
89	3	F3.2	Amplitude magnitude correction from station file. [JXCOR]
92	3	F3.2	Component correction for amp mag from XCM command. [ICOMX]
95	3	F3.2	Amplitude magnitude at this station. [KXMAG]
98	1	A1	Amp magnitude type code from XC1 or XC2 command. [LABX]
99	2	2X	Presently blank.

Duration magnitude data for the EARTHQUAKE

101	3	F3.2	Median event duration magnitude #1. [FMAG1]
104	1	A1	Type label for duration magnitude #1. [LABF1]
105	1	A1	Most common FMAG1 data source code. [FMSOU]
106	3	F3.2	Median-absolute-difference of coda magnitudes #1. [KFMMAD]
109	4	F4.1	Total of FMAG1 weights (i.e. number of readings). [MTFMAG]
113	1	1X	Presently blank.

114	3	F3.2	Median event duration magnitude #2. [FMAG2]
117	1	A1	Type label for duration magnitude #2. [LABF2]
118	1	A1	Most common FMAG2 data source code. [FMSOU2]
119	3	F3.2	Median-absolute-difference of coda mags #2. [KFMMAD2]
122	4	F4.1	Total of FMAG2 weights (i.e. number of readings). [MTFMAG2]
126	1	1X	Presently blank.

Amplitude magnitude data for the EARTHQUAKE

127	3	F3.2	Median event amplitude magnitude #1. [XMAG1]
130	1	A1	Type label for amplitude magnitude #1. [LABX1]
131	1	A1	Most common XMAG1 data source code. [XMSOU]
132	3	F3.2	Median-absolute-difference of amp magnitudes #1. [KXMMAD]
135	4	F4.1	Total of XMAG1 weights (i.e. number of readings). [MTXMAG]
139	1	1X	Presently blank.

140	3	F3.2	Median event amplitude magnitude #2. [XMAG2]
143	1	A1	Type label for amplitude magnitude #2. [LABX2]
144	1	A1	Most common XMAG2 data source code. [XMSOU2]
145	3	F3.2	Median-absolute-difference of amp magnitudes #2. [KXMMAD2]
148	4	F4.1	Total of XMAG2 weights (i.e. number of readings). [MTXMAG2]
152	1	1X	Presently blank.

Preferred magnitude data for the EARTHQUAKE

153	3	F3.2	Preferred magnitude. [PMAG]
156	1	A1	Type label for preferred magnitude. [LABPR]
157	3	F3.2	Median-absolute-difference of preferred mag. [KPRMAD]
160	4	F4.1	Total of preferred mag weights (i.e. number of readings).
164	1	1X	Presently blank.

External magnitude data for the EARTHQUAKE

165	3	F3.2	External magnitude. [BMAG]

| 168 | 1 | A1 | Type label for external magnitude. [BMTYP] |
| 169 | 3 | F3.1 | Total of preferred mag weights (i.e. number of readings). |

Quality codes (Hypo71 summary output only)

A 1-letter quality code is output on Hypo71 summary lines for consistency with this older program. Hypoinverse format has never calculated this code because it is an oversimplified parameter and is often misused. Seismologists would be better to use the individual criteria (number of stations, location errors, RMS, etc) rather than a simplified A-D rating. For example, an epicenter outside the network might be well controlled with S arrivals, but have a large DMIN or a large depth error and thus get a low rating, even if epicenter control is what is most important.

Two quality ratings are computed, and they are averaged to one for output. Tests are based on several parameters: RMS (root-mean-square travel-time residual), ERH (horizontal location error), ERZ (vertical location error), NWR number of weighted station readings (phases), MAXGAP (maximum angular gap in degrees between azimuthally adjacent stations), DEPTH (the earthquake depth), and DMIN (distance to closest station).

The first quality rating is based on errors and goodness-of-fit:

A. RMS < 0.15 sec <u>and</u> ERH $\leq$ 1.0 km <u>and</u> ERZ $\leq$ 2.0 km

B. RMS < 0.30 sec <u>and</u> ERH $\leq$ 2.5 km <u>and</u> ERZ $\leq$ 5.0 km

C. RMS < 0.50 sec <u>and</u> ERH $\leq$ 5.0 km

D. Worse than above

The second quality rating is based on station geometry:

A. NWR $\geq$ 6 <u>and</u> MAXGAP $\leq$ 90 <u>and</u> either DMIN $\leq$ DEPTH <u>or</u> DMIN $\leq$ 5.0

B. NWR $\geq$ 6 <u>and</u> MAXGAP $\leq$ 135 <u>and</u> either DMIN $\leq$ 2*DEPTH <u>or</u> DMIN $\leq$ 10

C. NWR $\geq$ 6 <u>and</u> MAXGAP $\leq$ 180 <u>and</u> DMIN $\leq$ 50

D. Worse than above

OLD PRE-Y2000 OUTPUT FORMATS

Recall that the "200" command selects between the old non-Y2000 formats and the current Y2000 formats with 4-digit years.

Pre-Y2000 summary format (also used as header in pre-Y2000 archive file)

Start Col.	Len.	Fortran Format	Data
1	10	5I2	Year, month, day, hour and minute.
11	4	F4.2	Origin time seconds.
15	2	F2.0	Latitude (deg).
17	1	A1	S for south, blank otherwise.
18	4	F4.2	Latitude (min).

22	3	F3.0	Longitude (deg).
25	1	A1	E for east, blank otherwise.
26	4	F4.2	Longitude (min).
30	5	F5.2	Depth (km).
35	2	F2.1	Amplitude magnitude.
37	3	I3	Number of P & S times with final weights greater than 0.1.
40	3	I3	Maximum azimuthal gap.
43	3	F3.0	Distance to nearest station (km).
46	4	F4.2	RMS travel time residual (sec).
50	3	F3.0	Azimuth of largest principal error (deg E of N).
53	2	F2.0	Dip of largest principal error (deg).
55	4	F4.2	Size of largest principal error (km).
59	3	F3.0	Azimuth of intermediate principal error.
62	2	F2.0	Dip of intermediate principal error.
64	4	F4.2	Size of intermediate principal error (km).
68	2	F2.1	Coda duration magnitude.
70	3	A3	Event location remark (region), derived from location.
73	4	F4.2	Size of smallest principal error (km).
77	1	A1	Auxiliary remark from analyst (i.e. Q for quarry).
78	1	A1	Auxiliary remark from program (i.e. "-" for depth fixed, etc.).
79	2	I2	Number of S times with weights greater than 0.1.
81	4	F4.2	Horizontal error (km).
85	4	F4.2	Vertical error (km).
89	2	I2	Number of P first motions.
91	3	F3.1	Total of amplitude magnitude weights (~ number of readings).
94	3	F3.1	Total of duration magnitude weights (~ number of readings).
97	3	F3.2	Median-absolute-difference of max amplitude magnitudes.
100	3	F3.2	Median-absolute-difference of duration magnitudes.
103	2	A3	3-letter code of crust and delay model.
106	1	A1	"Authority" code, i.e. what network furnished the information. Hypoinverse passes this code through.
107	1	A1	Most common P & S data source code.
108	1	A1	Most common duration data source code.
109	1	A1	Most common amplitude data source code.
110	1	A1	Primary coda duration magnitude type code
111	3	I3	Number of valid P & S readings (assigned weight > 0).
114	1	A1	Primary amplitude magnitude type code
115	1	A1	External magnitude label or type code. Hypoinverse pass-thru.
116	3	F3.2	External magnitude.
119	3	F3.1	Total of the "external" magnitude weights (~ number of readings).
122	1	A1	Alternate amplitude magnitude label or type code.
123	3	F3.2	Alternate amplitude magnitude.
126	3	F3.1	Total of the alternate amplitude mag weights.
129	10	I10	Event identification number
139	1	A1	Preferred magnitude label code chosen from those available.
140	3	F3.2	Preferred magnitude, chosen by the Hypoinverse PRE command.
143	3	F3.1	Total of the preferred mag weights (~ number of readings).
146	1	A1	Alternate coda duration magnitude label or type code.
147	3	F3.2	Alternate coda duration magnitude.

150	3	F3.1	Total of the alternate coda duration magnitude weights (~ number of readings).
153	1	A1	"Version" of the information, i.e. the stage of processing. This can either be passed through, or assigned by Hypoinverse with the LAB command.
154	1	A1	Version of last human review. Hypoinverse passes this through.

Pre-Y2000 Hypo71 summary output format

Start Col.	Len.	Fortran Format	Data (* revised from original format)
1	6	3I2	Year, month and day.
7	1	1X	Must be blank
8	4	2I2	Hour and minute.
12	6	F6.2	Origin time seconds.
18	3	F3.0	Latitude (deg).
21	1	A1	S for south, blank otherwise.
22	5	F5.2	Latitude (min).
27	4	F4.0	Longitude (deg).
31	1	A1	E for east, blank otherwise.
32	5	F5.2	Longitude (min).
37	7 (8)	F7.2, 1X	Depth (km).
45	1	A1	Preferred magnitude label code. *
46	5	F5.2	Preferred magnitude. *
51	3	I3	Number of P & S times with weights greater than 0.1.
54	4	F4.0	Maximum azimuthal gap.
58	5	F5.1	Distance to nearest station (km).
63	5	F5.2	RMS travel time residual.
68	5	F5.1	Horizontal error (km).
73	5	F5.1	Vertical error (km).
78	1	A1	Remark assigned by analyst (i.e. Q for quarry blast).
79	1	A1	Quality code A-D. See note above.
80	1	A1	Most common data source code. (i.e. W= earthworm).*
81	1	A1	Auxiliary remark from program (i.e. "-" for depth fixed, etc.).*
82	10	I10	Event identification number.*
92	1	A1	"Version" of the information, i.e. the stage of processing. This can either be passed through, or assigned by Hypoinverse with the LAB command.* 92 is the last filled column.

Pre-Y2000 station archive format

The archive format contains essentially all of the data about the event. It has enough data to relocate the event, and archive files can thus be both input and output. Archive output can be reformatted by programs like ARCPRINT to look like the print file, but the archive file does not contain the iteration history. The archive file may be read by other programs to plot first motions on the focal sphere (FPFIT, FPPLOT), compile residual summaries, extract a subset of events (SELECT, EXTRACT), etc.

The first line of each event in an archive file is identical to a HYPOINVERSE summary line and acts as a header. Next is one line per station, and a terminator line terminates each event. If the archive file has shadow records, a shadow record beginning with a "$" follows every header, station and terminator line.

For historical reasons, many fields are in the same position as the traditional phase input format. In fact, if you accidentally read an archive file expecting phase format (COP 1), you will get a location but it will be incomplete and subtle errors may result.

Start Col.	Len.	Fortran Format	Data
1	4	A4	4-letter station site code.
5	2	A2	P remark such as "IP".
7	1	A1	P first motion.
8	1	I1	Assigned P weight code (0-9 or blank).
9	1	A1	One letter station component code.
10	10	5I2	Year, month, day, hour and minute.
20	5	F5.2	Second of P arrival.
25	4	F4.2	P travel time residual.
29	3	F3.2	P weight actually used.
32	5	F5.2	Second of S arrival.
37	2 (3)	A2, 1X	S remark such as "ES".
40	1	I1	Assigned S weight code.
41	4	F4.2	S travel time residual.
45	3	F3.0	Peak-to-peak amplitude in Develocorder or WA mm.
48	3	F3.2	S weight actually used.
51	4	F4.2	P delay time.
55	4	F4.2	S delay time.
59	4	F4.1	Epicentral distance (km).
63	3	F3.0	Emergence angle at source.
66	1	I1	Amplitude magnitude weight code.
67	1	I1	Duration magnitude weight code.
68	3	F3.2	Period at which the amplitude was measured for this station.
71	1	A1	1-letter station remark.
72	4	F4.0	Coda duration in seconds.
76	3	F3.0	Azimuth to station in degrees E of N.
79	2	F2.1	Duration magnitude for this station.
81	2	F2.1	Amplitude magnitude for this station.
83	4	F4.3	Importance of P arrival.
87	4	F4.3	Importance of S arrival.
91	1	1X	Presently blank.
92	1	A1	Data source code.
93	1	A1	Label code for duration magnitude from FC1 or FC2 command.
94	1	A1	Label code for amplitude magnitude from XC1 or XC2 command.
95	1	A1	5th letter of station site code (optional).
96	3	A3	3-letter station component code.
99	2	A2	2-letter seismic network code.
101	2	A2	2-letter station location code (component extension).
			102 is the last filled column.

Y2000 OUTPUT FORMATS

Recall that the "200" command selects between the old non-Y2000 formats and the current Y2000 formats. The Y2000 formats have 4-digit years.

Y2000 summary format (also used as header in archive file)

Start Col.	Len.	Fortran Format	Data (* revised from pre-Y2000 format)
1	4	I4	Year. *
5	8	4I2	Month, day, hour and minute.
13	4	F4.2	Origin time seconds.
17	2	F2.0	Latitude (deg). First character must not be blank.
19	1	A1	S for south, blank otherwise.
20	4	F4.2	Latitude (min).
24	3	F3.0	Longitude (deg).
27	1	A1	E for east, blank otherwise.
28	4	F4.2	Longitude (min).
32	5	F5.2	Depth (km).
37	3	F3.2	Amplitude magnitude. *
40	3	I3	Number of P & S times with final weights greater than 0.1.
43	3	I3	Maximum azimuthal gap, degrees.
46	3	F3.0	Distance to nearest station (km).
49	4	F4.2	RMS travel time residual.
53	3	F3.0	Azimuth of largest principal error (deg E of N).
56	2	F2.0	Dip of largest principal error (deg).
58	4	F4.2	Size of largest principal error (km).
62	3	F3.0	Azimuth of intermediate principal error.
65	2	F2.0	Dip of intermediate principal error.
67	4	F4.2	Size of intermediate principal error (km).
71	3	F3.2	Coda duration magnitude. *
74	3	A3	Event location remark (region), derived from location.
77	4	F4.2	Size of smallest principal error (km).
81	1	A1	Auxiliary remark from analyst (i.e. Q for quarry).
82	1	A1	Auxiliary remark from program (i.e. "-" for depth fixed, etc.).
83	3	I3	Number of S times with weights greater than 0.1.
86	4	F4.2	Horizontal error (km).
90	4	F4.2	Vertical error (km).
94	3	I3	Number of P first motions. *
97	4	F4.1	Total of amplitude mag weights ~number of readings.*
101	4	F4.1	Total of duration mag weights ~number of readings. *
105	3	F3.2	Median-absolute-difference of amplitude magnitudes.
108	3	F3.2	Median-absolute-difference of duration magnitudes.
111	3	A3	3-letter code of crust and delay model.
114	1	A1	"Authority" code, i.e. what network furnished the information. Hypoinverse passes this code through.
115	1	A1	Most common P & S data source code. (See table 1 below).
116	1	A1	Most common duration data source code. (See cols. 68-69)
117	1	A1	Most common amplitude data source code.
118	1	A1	Primary coda duration magnitude type code
119	3	I3	Number of valid P & S readings (assigned weight > 0).
122	1	A1	Primary amplitude magnitude type code
123	1	A1	"External" magnitude label or type code. Typically "L" (=ML) This information is not computed by Hypoinverse, but passed along, as computed by UCB.
124	3	F3.2	"External" magnitude.
127	3	F3.1	Total of "external" magnitude weights (~ number of readings).
130	1	A1	Alternate amplitude magnitude label or type code.
131	3	F3.2	Alternate amplitude magnitude.
134	3	F3.1	Total of the alternate amplitude mag weights ~no. of readings.

Start Col.	Len.	Fortran Format	Data (* revised from pre-Y2000 format)
137	10	I10	Event identification number
147	1	A1	Preferred magnitude label code chosen from those available.
148	3	F3.2	Preferred magnitude, chosen by the Hypoinverse PRE command.
151	4	F4.1	Total of the preferred mag weights (~ number of readings). *
155	1	A1	Alternate coda duration magnitude label or type code.
156	3	F3.2	Alternate coda duration magnitude.
159	4	F4.1	Total of the alternate coda duration magnitude weights. *
163	1	A1	"Version" of the information, i.e. the stage of processing. This can either be passed through, or assigned by Hypoinverse with the LAB command.
164	1	A1	Version of last human review. Hypoinverse passes this through. 164 is the last filled column.

Y2000 Hypo71 summary output format

Start Col.	Len.	Fortran Format	Data (* revised from pre-Y2000 format)
1	4	I4	Year *
5	4	2I2	Month and day.
9	1	1X	Must be blank
10	4	2I2	Hour and minute.
14	6	F6.2	Origin time seconds.
20	3	F3.0	Latitude (deg).
23	1	A1	S for south, blank otherwise.
24	5	F5.2	Latitude (min).
29	4	F4.0	Longitude (deg).
33	1	A1	E for east, blank otherwise.
34	5	F5.2	Longitude (min).
39	7	F7.2	Depth (km).
46	1	1X	Blank
47	1	A1	Preferred magnitude type code.
48	5	F5.2	Preferred magnitude.
53	3	I3	Number of P & S times with weights greater than 0.1.
56	4	F4.0	Maximum azimuthal gap.
60	5	F5.1	Distance to nearest station (km).
65	5	F5.2	RMS travel time residual.
70	5	F5.1	Horizontal error (km).
75	5	F5.1	Vertical error (km).
80	1	A1	Remark assigned by analyst (i.e. Q for quarry blast).
81	1	A1	Quality code A-D. See note above.
82	1	A1	Most common data source code. (i.e. W= earthworm).
83	1	A1	Auxiliary remark from program (i.e. "-" for depth fixed, etc.).
84	10	I10	Event identification number
94	1	1X	Blank
95	1	A1	"Version" of the information, i.e. the stage of processing. This can either be passed through, or assigned by Hypoinverse with the LAB command.
96	3	A3	3-letter location remark. * 98 is the last filled column.

Y2000 (station) archive format

Start Col.	Len.	Fortran Format	Data (* revised from pre-Y2000 format)
1	5	A5	5-letter station site code, left justified. *
6	2	A2	2-letter seismic network code. *

8	1	1X	Blank *
9	1	A1	One letter station component code.
10	3	A3	3-letter station component code. *
13	1	1X	Blank *
14	2	A2	P remark such as "IP".
16	1	A1	P first motion.
17	1	I1	Assigned P weight code.
18	4	I4	Year. *
22	8	4I2	Month, day, hour and minute.
30	5	F5.2	Second of P arrival.
35	4	F4.2	P travel time residual.
39	3	F3.2	P weight actually used.
42	5	F5.2	Second of S arrival.
47	2	A2	S remark such as "ES".
49	1	1X	Blank
50	1	I1	Assigned S weight code.
51	4	F4.2	S travel time residual.
55	7	F7.2	Amplitude (Normally peak-to-peak). *
62	2	I2	Amp units code. 0=PP mm, 1=0 to peak mm (UCB), 2=digital counts. *
64	3	F3.2	S weight actually used.
67	4	F4.2	P delay time.
71	4	F4.2	S delay time.
75	4	F4.1	Epicentral distance (km).
79	3	F3.0	Emergence angle at source.
82	1	I1	Amplitude magnitude weight code.
83	1	I1	Duration magnitude weight code.
84	3	F3.2	Period at which the amplitude was measured for this station.
87	1	A1	1-letter station remark.
88	4	F4.0	Coda duration in seconds.
92	3	F3.0	Azimuth to station in degrees E of N.
95	3	F3.2	Duration magnitude for this station. *
98	3	F3.2	Amplitude magnitude for this station. *
101	4	F4.3	Importance of P arrival.
105	4	F4.3	Importance of S arrival.
109	1	A1	Data source code.
110	1	A1	Label code for duration magnitude from FC1 or FC2 command.
111	1	A1	Label code for amplitude magnitude from XC1 or XC2 command.
112	2	A2	2-letter station location code (component extension).
			113 is the last filled column.

SAMPLE INPUT AND OUTPUT FILES

The sample event is a small Parkfield earthquake, detected and located in the regular NCSN processing. Because NCSN processing has many large input files that are used regularly, we use binary files and a startup file identified by the environment variable HYPINITFILE (under unix systems) to be run by the INI command. Samples of some of the original ascii files are listed here, rather than supplying full-sized files to do complete processing. Files, parameters and even program capabilities are always changing, so these files are merely a sample.

The basic control file of Hypoinverse commands that is invoked by the INI command in Menlo Park is cal2000.hyp. If a line starts with "*", Hypoinverse interprets it as a comment. Each Hypoinverse 3-letter command is followed by the necessary parameters. The fortran parser

ignores everything after the "/" on a line with free-format parameters, so these are also comments.

HYP COMMAND FILE

```
* Hypoinverse startup file for NCSN for newer 2000 formats
* To run this file with the hyp command "ini" put a statement like this in your
* .cshrc file: setenv HYPINITFILE /home/calnet/klein/hypfiles/cal2000.hyp
*
* "home" paths for hypfiles directory:
* puna:    /home/calnet/klein/hypfiles/
* swave:  /home1/calnet/klein/hypfiles/
* ebird:  /ebird/calnet/klein/hypfiles/
* hollyhock: /we/calnet/klein/hypfiles/
*
* STATION DATA
200 T 1900 0                        /Enable yr 2000 formats
LET 5 2 3                           /Use new, longer station codes
******* YOU MUST READ STATIONS BEFORE USING THE ATE, DEL, FMC OR XMC COMMANDS.
RST '/home/calnet/klein/hypfiles/allsta2.bin'  /Read binary file of all stations.
*H71 1 1 3                              /use new station file format
*STA '/home/calnet/klein/hypfiles/all2.sta'    /Read ASCII file of all stations.
*
XMC '/home/calnet/klein/hypfiles/all2000.xmc' T 0 /Read XMAG cors, use all data
FMC '/home/calnet/klein/hypfiles/all2000.fmc' T 0 /Read FMAG cors, use all data
UNK 6 'IRG1' 'IRG2' 'IRG3' 'IRGE' 'WWVB' 'IRG '          /CUSP phony station codes
*
* MULTIPLE CRUSTAL MODELS
RCR '/home/calnet/klein/hypfiles/multmod2.bin' /Read all model info in binary
* READ MODELS AND DELAYS IN ASCII:
*@/home/calnet/klein/hypfiles/multmod2.hyp
*
* MAGNITUDE CHOICES
MAG 1 T 3 1                         /Use Eaton for Fmag1, Hirshorn for Fmag2
DUR -.81 2.22 0 .0011 0, 5*0, 9999 1    /Set Eaton's new magnitude constants
DU2 .005 40, .0006 350, .014 10         /Extra dist & depth terms for Eaton
DUB -2.06 2.95 0 .001 0, 5*0, 9999 1    /Zmag of Hirshorn; use -2.06=-.71-1.35
* -1.35 compensates for the 1.35 gain corr used for the 42db Z stations
FC1 'D' 4 'VHZ' 'VHE' 'VHN' 'VLZ'        /comps for FMAG1, NEW CODE D
FC2 'Z' 1 'VLZ'                          /Use lowgain verts for FMAG2=Zmag
XC1 'X' 7 'VHZ' 'VLZ' 'VLE' 'VLN' 'VDZ' 'VDN' 'VDE' /XMAG components & label
XC2 'L' 6 'WLN' 'WLE' 'HHN' 'HHE' 'BHN' 'BHE' /LOCAL MAG (ML) comps
* Use BKY's Nordquist logA0 relation for the WA & synthetic WA components
LA0 6 'WLN' 4, 'WLE' 4, 'BHN' 4, 'BHE' 4, 'HHN' 4, 'HHE' 4
FCM 3 'VLZ' -.06 'VLE' -.30, 'VLN' -.30 /Component corrections for Eaton Fmags
XCM 2 'VHZ'  .33, 'VLZ'  .20          /Component corrections for Eaton Xmags
PRE 6, 3 0 4 9, 1 1 0 9, 2 1 0 9, 4 4 4 9, 3 0 0 9, 4 0 0 9 /Preferred mags
* Preferred mag, min readings, mag range. In pref order.
*
* STANDARD CHOICES FOR CALNET
RMS 4 .10 2 3                       /Residual weighting
ERR .10                             /Standard timing error
POS 1.78                            /P to S ratio
REP T F               /Log events to terminal; don't print unweighted stations
JUN T                               /Force location of junk events
MIN 4                               /Min number of stations
NET 2                               /Use California region codes
ZTR 5                               /Trial depth
DIS 4 15 3.5 7                      /Distance weighting
WET 1. .5 .2 .1                     /Weights for P weight codes 0-3
*
```

```
* OUTPUT FORMAT
ERF T                                   /Send error messages to terminal
TOP F                                   /No page ejects
LST 1 1 0                               /No station list or models in printfile
KPR 2                                   /Medium print output each event
*
* PMAG SETUP, 3-LETTER COMPONENT CODES, USE PMA, PMC, PC1, PC2 AND PRE COMMANDS
*PMA T T .04 .4 5    /TURN ON PMAG CALC & PRINTING, CNT2MM FACTOR, CLIPRATIO
*PMC 4 'W' .0488, 'P' .04, 'R' .04, 'O' .04 /DATA SOURCES W/ CNT2MM FACTORS
*PC1 'P' -1.418 1.760 -1       /PRIMARY PMAG FROM ALL COMPONENTS
*PC2 'G' 0.0 1.0 1 'VLZ'       /SECONDARY PMAG FROM LOW GAIN COMPONENT
*PRE 9, 28*, 7 2 0 9, 6 4 0 9    /Add pmags to preferred mag list
******* YOU SHOULD USE THE ATE COMMAND WITH THE DATE AND TIME OF YOUR FIRST EQ.
******* USING A YEAR OF 0 (THE ATE COMMAND BELOW NOT COMMENTED OUT) WILL ALWAYS
******* WORK, BUT IS INEFFICIENT.
*
******* AN EASY WAY IS TO USE THE ATE COMMAND BELOW WITH A 0 YEAR, THEN
******* FOLLOW THIS COMMAND FILE WITH YOUR OWN ATE COMMAND:
*******  ATE 1* 1984 1 1 0 0   THIS REREADS THE STANDARD FILE BUT LOADS
******* ATTENUATIONS FROM YOUR STARTING DATE.
*
*ATE '/home/calnet/klein/hypfiles/all2000.atn' 1984 1 1 0 0 /Put date of your first eq
ATE '/home/calnet/klein/hypfiles/all2000.atn' 0 /Use this if first date is unknown
CAL '/home/calnet/klein/hypfiles/all2000.cal' 0 /Load cal factors for digital stations
*
```

The above control file cal2000.hyp reads binary station and crust files to save time because there are so many large files. Here are a few lines of the station location file all2.sta that was read to make the binary file, as read with the STA command. It is in Hypoinverse station format #2:

STATION FILE FRAGMENT

```
PAB    NC VVHZ  35   9.4371 120 38.2559  1430.0   P 0.00   0.00   0.00   0.00 1  0.00
PABB   NC VVHZ  35   9.5352 120 38.3950  1700.0   P 0.00   0.00   0.00   0.00 1  0.00
PAD    NC VVHZ  35 38.4030 120 51.9729  4300.0   P 0.00   0.00   0.00   0.00 1  0.00
PAG    NC IASZ  35 43.9493 120 15.0209  4360.0   P 0.00   0.00   0.00   0.00 0  0.00
PAG    NC JASN  35 43.9493 120 15.0209  4360.0   P 0.00   0.00   0.00   0.00 0  0.00
PAG    NC KASE  35 43.9493 120 15.0209  4360.0   P 0.00   0.00   0.00   0.00 0  0.00
PAG    NC VVHZ  35 43.9493 120 15.0209  4360.0   P 0.00   0.00   0.00   0.00 1  0.00
PAN    NC VVHZ  35 46.8034 120 54.4279  4260.0   P 0.00   0.00   0.00   0.00 1  0.00
PAP    NC VVHZ  35 53.7514 121 22.0113 10440.0   P 0.00   0.00   0.00   0.00 1  0.00
PAR    NC VVHZ  36 14.9567 120 20.5662  4520.0   P 0.00   0.00   0.00   0.00 1  0.00
PAV    NC VVHZ  35 10.5500 120 37.9500  1330.0   P 0.00   0.00   0.00   0.00 1  0.00
```

A sample of the coda magnitude correction file read with the FMC command follows. It could be a history file, but in this case the corrections are assumed constant in time and their expiration dates are 0 (blank) indicating no expiration.

CODA MAGNITUDE CORRECTION FILE FRAGMENT

```
HCA    NC  VHZ  0.04
HCB    NC  VHZ -0.24
HCO    NC  VHZ -0.53
HCO    NC  VLZ -0.53
HCP    NC  VHZ -0.09
HCR    NC  VHZ  0.28
HDL    NC  VHZ  0.22
```

This is a sample of a layered crust model file read with the CRH command:

LAYERED CRUST MODEL FILE

```
PARKFIELD MODEL
1.42  0.00
3.24  0.25
4.82  1.50
5.36  2.50
5.60  3.50
5.87  6.00
6.15  9.00
6.60 15.00
8.00 25.00
```

A station delay file read with the DEL command (in this case for one particular model) looks like this:

STATION DELAY FILE FRAGMENT

```
PAG   NC  0.10
PAN   NC -0.20
PAP   NC -0.50
PAR   NC  1.25
PAV   NC  0.16
PBI   NC  0.33
PBM   NC -0.26
PBP   NC  0.25
PBR   NC -0.22
PBW   NC -0.13
PBY   NC -0.32
PCA   NC  0.72
```

This is a sample of an attenuation history file read with the ATE command. Calibration files are similar.

ATTENUATION HISTORY FILE FRAGMENT

```
ABL   CI   VHZ   6 1984103123 12 1985040120 18
ARV   CI   VHZ 12
BCH   CI   VHZ   6 1985092400 12
BMT   CI   VHZ   6
CRG   CI   VHZ 12
ECF   CI   VHZ 18
FTC   CI   VHZ 18
LJB   CI   VLZ 48
MAR   CI   VHZ 12
PKM   CI   VHZ   6 1985040120 12
```

Here is the command dialog (in a terminal window) that located the sample event. The user commands and responses that were typed at the keyboard are in **bold**.

COMMAND DIALOG

```
puna 45? hyp
HYPOINVERSE 2000 STARTING
 COMMAND? ini
 INITIALIZING WITH COMMAND FILE:
 /home/calnet/klein/hypfiles/cal2000.hyp

  3473 STATIONS READ IN BINARY
   540 STATION XMAG CORRECTIONS SET
   511 STATION FMAG CORRECTIONS SET
    35 CRUST MODELS READ IN BINARY
   804 STATION ATTENUATIONS SET
    53 STATION CAL FACTORS SET
 COMMAND? phs
 PHASE FILENAME
 [ CR = ]? testin.arc
 COMMAND? fil
   FIND INPUT PHASE FILE TYPE & SET PHS(COP) & ARC(CAR) FORMATS
   INPUT IS A HYPOINVERSE ARCHIVE-2000 FILE, NO SHADOWS
   SETTING FORMATS COP 3, CAR 1
 COMMAND? prt
PRINTOUT FILE (NONE FOR NONE)
 [ CR = ]? test.prt
 COMMAND? arc
 ARCHIVE FILE (NONE FOR NONE)
 [ CR = ]? test.arc
 COMMAND? loc
   SEQ ---DATE--- TIME REMARK -LAT-  --LON-  DEPTH  RMS PMAG  NUM ERH ERZ  ID
     1 1999-12- 1  0:08 MID   35 58  120 31   2.39 0.03 1.0D   8 1.1 3.1 21069577
 COMMAND? sto
puna 46?
```

The input phase file set with the PHS command for this event is in the archive-2000 format. The header line was not written by Hypoinverse because all fields are not filled in.

INPUT PHASE FILE

```
19991201 0 8 64735 5775120 3094  355  0                             160
PST  NC VVHZ IPD0199912010008 7.56                00 0                  4       0      0
PMM  NC VVHZ EPU2199912010008 7.71               700 0                  4 19M    0      0
PVC  NC VVHZ IPD1199912010008 7.81      8.98ESD2  00 0                  0       6      0
PPC  NC VVHZ IPU0199912010008 8.13      9.19ESU2 1600 0                  4  7M    0      0
PHP  NC VVHZ EPU2199912010008 8.39                00 0                  4       0      0
PSM  NC VVHZ EPU2199912010008 9.98                00 0                  4       0      0
BMS  NC VVHZ  P 419991201000899.99               00 0                  4      22      0
                                                            21069577
```

The output print file (set with the PRT command) has some of the basic program test parameters listed before the earthquake output:

OUTPUT PRINT FILE

```
 HYPOINVERSE 2000 (11/99 VERSION) RUN ON Mon Apr 24 16:31:42 2000     RUN LABEL=


 TEST PARAMETERS:
   -ITERATION AND CONVERGENCE-    -WEIGHTING AND ERRORS-     -MISCELLANEOUS-
      20=ITRLIM   0.900=DAMP   15.000=DISCUT  0.100=RMSCUT      4=MINSTA
    0.040=DQUIT   0.001=DRQT    3.500=DISW1   2.000=RMSW1       2=NET
    7.000=DXFIX   0.012=EIGTOL  7.000=DISW2   3.000=RMSW2
   30.000=DZMAX   0.020=RBACK       4=ITRDIS      4=ITRRES  5.000=ZTR
    0.500=DZAIR   0.600=BACFAC  1.000=SWT     0.100=RDERR   1.780=POS
                                              1.000=ERCOF
```

```
    ------DURATION MAG CONSTANTS------        -DELAYS & MISC-    -STATIONS-
  -0.810=FMA1      0.000=FMA2   -1.0300=DMA0        T=LMULT        F=ATTEN
   2.220=FMB1      0.000=FMB2    2.1000=DMA1        T=CODAWT       5=SITE-LET
   0.000=FMZ1      0.000=FMZ2    0.0000=DMA2        T=LJUNK        2=NET-LET
   0.001=FMD1      0.000=FMD2    0.0000=DMZ                        3=COMP-LET
   0.000=FMF1      0.000=FMF2    1.0000=DMGN
   1.000=FMGN   9999.000=FMBRK   0.0027=DMLIN
  0.0050=DCOF1    40.000=DBRK1        1=LOGA0
  0.0006=DCOF2   350.000=DBRK2
  0.0140=ZCOF     10.000=ZBRK

   DUR MAG COMPONENT CORRECTIONS:  VLZ=-.06  VLE=-.30  VLN=-.30
   AMP MAG COMPONENT CORRECTIONS:  VHZ=0.33  VLZ=0.20

   --- MAGNITUDE LABELS & COMPONENTS ---
   FMAG1: LABEL=D       COMPS= VHZ VHE VHN VLZ
   FMAG2: LABEL=Z       COMPS= VLZ
   XMAG1: LABEL=X       COMPS= VHZ VLZ VLE VLN VDZ VDN VDE
   XMAG2: LABEL=L       COMPS= WLN WLE HHN HHE BHN BHE
   FMAG1 MAGSEL=1  FMAG2 MAGSEL=3

INPUT FILES:
COMMANDS:    /home/calnet/klein/hypfiles/cal2000.hyp

BINARY STATION SNAPSHOT FILE: /home/calnet/klein/hypfiles/allsta2.bin
DELAYS:
ATTENUATIONS: /home/calnet/klein/hypfiles/all2000.atn
CAL FACTORS:  /home/calnet/klein/hypfiles/all2000.cal
FMAG CORRECT: /home/calnet/klein/hypfiles/all2000.fmc
XMAG CORRECT: /home/calnet/klein/hypfiles/all2000.xmc

PHASES:      testin.arc
       PHASE FORMAT CODE= 3, TERMINATOR FORMAT CODE=1
BINARY CRUST SNAPSHOT FILE: /home/calnet/klein/hypfiles/multmod2.bin

OUTPUT FILES: (T IF APPENDED TO)
(F) PRINTOUT:   test.prt
(F) ARCHIVE:    test.arc                                 ARCHIVE FORMAT CODE= 1

################################################################################
 1 DEC 1999,  0:08  SEQUENCE NO.   1, ID NO.   21069577
                                        ADJUSTMENTS (KM)
  I ORIGIN   LAT N      LON W     Z    NWR RMS    DT    DLAT    DLON    DZ   RR NF MOD
  1  5.56  35 56.14  120 31.19  5.00    8 0.94  0.817   2.645  -0.440   0.000 2.681 3 PMM
FREE DEPTH
  2  6.38  35 57.57  120 30.90  5.00    8 0.05  0.170   0.539   0.233  -1.916 2.004 4 PMM
  3  6.55  35 57.86  120 31.05  3.08    8 0.07  0.003  -0.156   0.177  -0.585 0.631 4 PMM
BEGIN DISTANCE WEIGHTING
BEGIN RESIDUAL WEIGHTING
  4  6.55  35 57.77  120 31.17  2.50    8 0.03 -0.007  -0.038   0.092  -0.189 0.214 4 PMM
  5  6.54  35 57.75  120 31.23  2.31    8 0.03  0.003  -0.037  -0.003   0.082 0.090 4 PMM
  6  6.55  35 57.73  120 31.23  2.39    8 0.03 -0.001   0.015  -0.042   0.084 0.095 4 PMM
ERROR ELLIPSE: <SERR AZ DIP>-<  3.28  86 69>-<  0.58 213 12>-<  0.21 308 15>
--------------------------------------------------------------------------------
YEAR MO DA  --ORIGIN--  --LAT N-  --LON W--  DEPTH   RMS   ERH   ERZ  XMAG  FMAG  PMAG
1999-12-01  0008  6.55  35 57.73  120 31.23   2.39  0.03  1.14  3.08  0.56  0.96  0.96D
                                                                               SOURCE
NSTA NPHS  DMIN MODEL GAP ITR NFM NWR NWS NVR REMRKS-AVH  N.XMG-XMMAD-T  N.FMG-FMMAD-T  L F X
   7    9   2.1  PMM  137   6   6   8   2   8 MID          2.00 0.14 X    1.00 0.00 D   N N N

XMAG2-N.XMG2-XMMAD-T-S  FMAG2-N.FMG2-FMMAD-T-S  PREF.MAG-N.PMAG-PRMAD-T
                                                  0.96  1.00  0.00 D
REGION= Middle Mountain             MODELS USED:  PMM=1.00

STA NET COM L CR DIST AZM  AN P/S WT   SEC (TOBS -TCAL -DLY =RES)  WT   SR  INFO  CAL DUR-W-FMAG-T -AMP-U-PER-W-XMAG-T
PMM  NC VHZ   VP 2.1 108 124 EPU 2  7.71  1.16  0.89  0.29 -0.02  0.46  NM 0.612 1.40                7.000M .19  0.42 X
PST  NC VHZ   VP 3.5 171 102 IPD    7.56  1.01  1.16 -0.15  0.00  2.29  N  0.829
PVC  NC VHZ   VP 4.7 197  77 IPD 1  7.81  1.26  1.39 -0.09 -0.04  1.14  N  0.341 1.40  6    0.96 D
                            ES  2  8.98  2.43  2.47 -0.16  0.12  0.46S N  0.272
PPC  NC VHZ 01VP 6.9 259  72 IPU    8.13  1.58  1.81 -0.24  0.01  2.29  NM 0.860 1.40               16M .07  0.69 X
                            ES  2  9.19  2.64  3.22 -0.43 -0.15  0.46S NM 0.199
PHP  NC VHZ   VP 8.0 285  72 EPU 2  8.39  1.84  2.03 -0.21  0.02  0.46  N  0.080
```

```
    PSM  NC VHZ   VP 13.7 331  71 EPU 2   9.98  3.43  3.09 0.37 -0.03  0.46  N  0.804
      1 UNWEIGHTED STATIONS NOT PRINTED.
```

The first line of the archive file is the summary header. The archive output file is difficult (for people) to read, especially with the word wrapping of the summary line, which occurs because the line is long.

OUTPUT ARCHIVE FILE

```
199912010008065535 5773120 3123  239 56  8137  2   3 8669 32821312  58 96MID  21     2 114 308  6  20  10 14  0PMM NNND  8X
0  0L  0  0  21069577D 96  10Z  0   0
PMM  NC VVHZ EPU2199912010008  771  -2 46   0   0    07.00000 0  0  29   0  2112404 19M   0108  0 42 612   0N X
PST  NC VVHZ IPD0199912010008  756   0229   0   0   0    0 0  0 -15   0  3510204 0    0171  0  0 829   0N
PVC  NC VVHZ IPD1199912010008  781  -4114  898ES 2  12    0 0 46  -9 -16  47 7700 0    6197 96  0 341 272ND
PPC  NC VVHZ IPU0199912010008  813   1229  919ES 2 -1516.0000 0 46 -24 -43  69 7204 7M   0259  0 69 860 199N X
PHP  NC VVHZ EPU2199912010008  839   2 46   0   0   0    0 0  0 -21   0  80 7204 0    0285  0  0  80   0N
PSM  NC VVHZ EPU2199912010008  998  -3 46   0   0   0    0 0  0  37   0 137 7104 0    0331  0  0 804   0N
BMS  NC VVHZ  P 4199912010008 99997826  0   0   0   0    0 0  0  43   0 815 5504 0  22343211  0   0   0ND
                                                  21069577
```

APPENDIX

Appendix 1: Rules for free format parameters following commands

- Supply the parameters in free-format following the command.

- The type and order of parameters is the same as in the command documentation.

- Free-format values may be separated by either spaces, commas or tabs.

- Character strings (for filenames, labels etc.) are delimited by apostrophes like 'MYFILE.DAT' .

- The form "n*A" stands for n occurrences of the value A.

- A null field will leave the existing value unchanged. A null field is specified by two consecutive commas, by one leading comma or by two trailing commas.
  Thus, "2,, 'MYFILE.' ,," changes only the 1st and 3rd of 4 values.

- A slash (/) at the end of a line means all later fields are null.

- The form "n*" stands for n occurrences of a null field.

Appendix 2: Iteration and convergence

The solution to the earthquake location problem is non-linear. It is solved by a series of iterations in linear steps to converge on the minimum RMS (root-mean-square travel time residual), which is the best estimate of the hypocenter. The RMS is

$$RMS^2 = \frac{\sum (w_i r_i)^2}{\sum (w_i)^2}$$

where r_i are the individual station residuals, and w_i are their final weights. The location method involves 1) guessing a trial location, 2) calculating the RMS at the current location 3) calculating an adjustment vector in the direction which minimizes the RMS, 4) take that adjustment, or a modification of it, and 5) repeat steps 2-4 until the solution converges or meets some criteria.

Where to begin iterations

In the absence of a specified trial origin time, latitude, longitude, or depth on the terminator line, a standard trial hypocenter is assumed. (Any one of the four trial hypocenter parameters may be specified independently, however.) The trial origin time is two seconds before the first arrival, and the trial epicenter is under the station with the first arrival. Starting depth is at the trial depth ZTR specified with the ZTR command. During the early iterations (usually just the first), depth is held fixed until the horizontal adjustment is less than DXFIX specified with the DAM command. If the trial depth ZTR is negative, all events in this run are held fixed at this depth (at the positive value), unless ZTR is temporarily overridden by a trial depth which is set for a particular event on its terminator line.

How iterative steps may be modified

Various parameters can be defined which damp the epicentral adjustments if the adjustment vector becomes large or unstable. DAMP (DAM command) is the damping factor by which all hypocenter adjustments are always multiplied before an iterative step is taken. Damping is automatically increased by cutting DAMP in half for the last 1/3 of the allowed number of iterations. Thus, if 15 iterations are allowed and convergence has not been reached after 10 iterations, the remaining 5 iterations will be heavily damped. Empirically this appears to improve convergence.

If an iterative step would place the hypocenter in the air, the hypocenter is moved up to the fraction (l.-DZAIR) (DAM command) of its present depth. Earthquakes above the model surface are not allowed because a flat earth model is assumed with all stations at its surface. Thus the depth adjustment is -DZAIR * Z. The depth adjustment may be independently damped if the adjustment is larger than DZMAX (DAM command). If it is, the depth variation is damped by the factor DZMAX /(DZ + DZMAX) where DZ is the calculated depth adjustment.

If the value of RMS should increase by more than the amount RBACK (DAM command) after the last iteration, the hypocenter is moved back by the fraction BACFAC (DAM command) toward the previous hypocenter. This situation often occurs when a poorly constrained hypocenter iterates across a large velocity discontinuity in the crustal model.

The use of a generalized inverse scheme for finding the hypocenter adjustment allows great control over the adjustments actually taken. For example we may choose not to make hypocenter adjustments in directions which are poorly constrained by the arrival time data and which are directions in which location errors are large. The parameter EIGTOL (DAM command) does exactly this, and serves as a cut-off below which eigenvalues of the inversion are deemed unstable and are suppressed. See the appendix on the inversion scheme for a brief description of the matrix equations that might aid in using this parameter. Two other parameters are useful in curbing large and generally erratic steps when the solution is poorly constrained. Individual distance steps are limited to DXMAX (DAM command). When the distance to the second closest station exceeds D2FAR (for example, a large number like 250 km set with the DAM command), stop iterating because the earthquake is outside the network and has no control.

Convergence and when to stop iterating

Normally, iteration can stop in any of 3 ways: 1) when the number of iterations reaches the maximum allowed (ITRLIM, set with the CON command); 2) When the *change* in the RMS residual between iterations becomes less than DRQT (CON command); or 3) When the hypocenter adjustment vector is less than DQUI.T km (set with the CON command). The last two tests are only applied after a) the depth has been freed from its trial value for at least one iteration, and b) after residual and distance weighting have been applied.

Appendix 3: Inversion scheme and use of eigenvalue cutoff

In what follows, bold variables are vectors or matrices, and n or m are their dimensions. This derivation follows treatments by Geiger (1912), Eaton (1969) and Lawson and Hanson (1974). Let's first consider a simplified case. If the solution to the earthquake location problem were

linear and if we had exactly as many independent data m (arrivals times) as hypocentral unknowns n, the answer would be the solution of:

$$\mathbf{T} = \mathbf{A} \cdot \mathbf{X} + \mathbf{G}$$
$$\quad n \quad nxn \quad n \quad n$$

where $\mathbf{T}$ is the n-vector of arrival times, $\mathbf{X}$ is the n-vector of hypocenter coordinates and $\mathbf{G}$ is constant. $\mathbf{A}$ is the n by n partial derivative matrix

$$A_{ij} = \frac{\partial T_i}{\partial X_j}$$

that may be directly calculated from an assumed velocity model. But since the earthquake problem is nonlinear ($\mathbf{A}$ is not constant), we must seek successive linearized solutions and iterate toward the true solution until we have converged to the desired accuracy. $\mathbf{X}$ and $\mathbf{A}$ must also be updated as iteration proceeds. If $\mathbf{T_0}$ and $\mathbf{X_0}$ are the arrival time and hypocenter vectors calculated at the previous step (or some initial guess on the first iteration) which satisfy

$$\mathbf{T_0} = \mathbf{A} \cdot \mathbf{X_0} + \mathbf{G}$$

then subtracting the equations yields

$$\mathbf{T} - \mathbf{T_0} = \mathbf{A} \cdot (\mathbf{X} - \mathbf{X_0}), \text{ or } \mathbf{R} = \mathbf{A} \cdot \mathbf{DX}$$
$$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad n \quad nxn \quad n$$

where $\mathbf{R}$ is the vector of travel time residuals (observed times minus those calculated from the model at the previous step) and $\mathbf{DX}$ is the hypocentral adjustment vector, given in this simple case by $\mathbf{DX} = \mathbf{A}^{-1} \cdot \mathbf{R}$.

Now we must consider the real case. The number of observations m for the earthquake problem is often in the range 8 to 60, but the number of unknowns n is generally only 4. When m exceeds n, however, the true inverse $\mathbf{A}^{-1}$ does not exist. We seek the least squares solution which best solves

$$\mathbf{R} = \mathbf{A} \cdot \mathbf{DX}$$
$$\quad m \quad mxn \quad n$$

in the sense of minimizing

$$(\mathbf{R} - \mathbf{A} \cdot \mathbf{DX})^2$$

This is done by pre-multiplying by $\mathbf{A}^T$ (transpose) to get the least-square condition

$$\mathbf{A}^T \cdot \mathbf{R} = (\mathbf{A}^T \cdot \mathbf{A}) \cdot \mathbf{DX}$$
$$\quad nxm \quad m \quad nxn \quad n$$

that now only requires inversion of the nxn symmetric matrix $\mathbf{A^T \cdot A}$.

The solution can be sought in terms of the generalized inverse of $\mathbf{A}$, and in particular the singular value decomposition (SVD) of $\mathbf{A}$. This not only yields the usual least square solution, but permits manipulation of the eigenvalues of $\mathbf{A^T \cdot A}$, calculation of the errors, and evaluation of the information content of the data. Hypoinverse uses the SVD subroutine of Lawson and Hanson (1974) and forms the above matrices from elements of the decomposition.

The decomposition of $\mathbf{A}$ is given by

$$
\begin{array}{ccccccc}
\mathbf{A} & = & \mathbf{U} & \cdot & \mathbf{S} & \cdot & \mathbf{V^T} \\
\text{mxn} & & \text{mxn} & & \text{nxn} & & \text{nxn}
\end{array}
$$

where $\mathbf{U}$ and $\mathbf{V}$ are eigenvector matrices and $\mathbf{S}$ is the diagonal matrix of eigenvalues of $\mathbf{A^T \cdot A}$. Also

$$\mathbf{U^T U = I} \text{ and } \mathbf{V^T V = I},$$

($\mathbf{I}$ is the identity matrix) and assuming that the number of linearly independent arrival time data exceeds the number of unknowns, $\mathbf{VV^T = I}$. When the resolution matrix $\mathbf{VV^T}$ equals the identity matrix, the unknowns are perfectly resolved which is the usual case for the earthquake problem. Then the least-squares solution can be derived by substitution of $\mathbf{A}$ into the least-squares condition and is given by

$$
\begin{array}{ccccccccc}
\mathbf{DX} & = & \mathbf{V} & \cdot & \mathbf{S^{-1}} & \cdot & \mathbf{U^T} & \cdot & \mathbf{R} \\
\text{n} & & \text{nxn} & & \text{nxn} & & \text{nxm} & & \text{m}
\end{array}
$$

The covariance matrix of the solution $\mathbf{DX}$ is given by

$$
\begin{array}{ccccccc}
\mathbf{C} & = & w^2 & \mathbf{V} & \cdot & \mathbf{S^{-2}} & \cdot & \mathbf{V^T} \\
\text{nxn} & & & \text{nxn} & & \text{nxn} & & \text{nxn}
\end{array}
$$

where w is a constant.

We see at once that if one or more eigenvalues in $\mathbf{S}$ becomes small, both the solution vector and error become large and unstable. Each eigenvalue corresponds to one of the mutually orthogonal principal directions of the solution, and if one eigenvalue becomes small, both the adjustment DX_i and standard error $\sqrt{C}$ in that principal direction become large in proportion to one over that eigenvalue. The principal direction with the small eigenvalue will in general include components of origin time, latitude, longitude, and depth. Most often, however, the smallest eigenvalue has its largest component in depth. If an eigenvalue should become smaller than the parameter EIGTOL (set with the DAM command), no adjustment is taken in that principal direction for which the error is also large. The program does not add the term to $\mathbf{DX}$ originating from the small eigenvalue. In other words, solutions are prevented from becoming unstable and scattering out in the direction in which their error ellipsoids are very long.

In general, the largest eigenvalue is of order 5 (with its dominant component in origin time) and the spatial eigenvalues are of order 0.3 to 0.7. The difference in size between origin time and

spatial eigenvalues arises because a change of several km in hypocenter location is required to produce the same change in an arrival time as a one second change in origin time. Unstable or very poorly constrained situations occur when the smallest eigenvalue becomes less than about 0.02. Looked at another way, instability occurs when the condition number (ratio of largest to smallest eigenvalue) exceeds about 200. The value of EIGTOL should be chosen after attempting to solve for the most marginal events one wishes to locate with a given network, and studying the eigenvalues and iteration history for these events.

The station importance and information density

The station importance is a product of the generalized inverse approach and usually is not computed by other standard location programs. It is a quantitative measure of the contribution a particular arrival makes to the hypocenter solution, and includes the effect of the weights applied to the arrival data.

A result of the singular value decomposition of the partial derivative matrix **A** (see section on inversion scheme) is the information density matrix $\mathbf{B} = \mathbf{U} \cdot \mathbf{U}^T$. This is an m x m matrix, where m is the number of arrival times reported. Each diagonal element b_{ii} of **B** is thus associated with the ith phase alone, and is the quantity printed and referred to as the importance of the phase arrival.

A feeling for what importance means quantitatively may come from realizing that the rows of **U** are linearly related to the rows of the partial derivative matrix **A**. In other words, when the partial derivatives of the travel time to the ith station with respect to the jth hypocentral coordinate $\partial T_i/\partial X_j$ are large for the ith station (the ith row of **A**); then the ith row of **U**, and hence the station importance b_{ii}, will also be large. Thus a large leverage, through the partial derivative matrix **A**, of a particular station on the solution is equivalent to a large station importance. This can be seen intuitively from the relation:

```
A • V = U • S
mxn nxn mxn nxn
```

where the matrices are as defined in the inversion section. When the ith row of **A** is large (corresponding to the ith station), the ith row of this equation and hence of **U** will be large. The ith diagonal element s_{ii} of $\mathbf{U} \cdot \mathbf{U}^T$ will also be large.

An illustration of the relation between importance and partial derivatives is the fact that an S reading has a greater importance than a P reading from the same station. The partial derivatives ∂ (travel time)/ ∂ (space coordinate), are larger for S arrivals than P arrivals at the same station by the factor T_S/T_P, and this means that rows of the **U** matrix and consequently the importance will be larger for the S arrivals. This has an important consequence for assigning weights to arrivals. When an S arrival cannot be read to the same precision as a P arrival, it should be given less weight to compensate for its intrinsically larger importance.

The importance is a measure of the redundancy in the data, and for example is individually small in distances and azimuths where there are many stations. This can be seen from the following argument. The inversion process for the over-determined earthquake problem extracts n linearly independent combinations of partial derivatives from the m combinations in the matrix **A**. One "unit" of importance is attributed to each of these n independent combinations. Hence the sum of the importances of all stations for a full earthquake solution is 4. If several data are redundant,

i.e. linearly dependent or nearly so, then the unit of importance must be distributed among them and the importance of each redundant datum goes down.

Eigenvalue and covariance (error) output

If KPRINT is 3 or larger (KPR command), the four eigenvalues of the principal directions of the solution are listed in descending order. These are useful in gauging the relative stability and error of the solution in the four principal directions. Under each eigenvalue are the column eigenvectors corresponding to it. The eigenvectors together make up the matrix **V**. The elements of the column eigenvectors give the components of origin time, latitude, longitude and depth in the principal direction corresponding to that eigenvalue. In other words, the matrix of eigenvectors accomplishes the "rotation" between the principal and geographic coordinates. The last eigenvector gives the mix of latitude, longitude and depth that are most poorly determined and associated with the smallest eigenvalue.

The covariance matrix gives the variances (diagonal elements) and covariances of origin time, latitude, longitude and depth. The errors listed are the standard errors of origin time (in sec), and latitude, longitude and depth (in km) with the other three variables held fixed. They are the square roots of the diagonal elements of the covariance matrix. The error ellipsoid consists of the lengths of the principal axes SERR and their azimuths AZ and dips DIP in degrees. The principal axes are the standard errors in those directions in units of km. The hypocenter statistically has a 32% chance of lying within the error ellipsoid given. To obtain a 95% confidence ellipsoid, multiply the standard errors by 2.4. See the sections on the inversion scheme and error calculations for more information.

Appendix 4: Error calculations

The covariance or error matrix

The covariance or error matrix is calculated from elements of the decomposition of the **A** matrix (see section on inversion scheme) as

$$
\mathbf{C} = w^2 \quad \mathbf{V} \cdot \mathbf{S}^{-2} \cdot \mathbf{V^T}
$$
$$
\text{nxn} \qquad \text{nxn} \quad \text{nxn} \quad \text{nxn}
$$

where **S** and **V** are matrices composed of eigenvalues and eigenvectors in the "solution space" of the hypocenter. w^2 is the variance (standard error squared) of the arrival time data. Hypoinverse calculates w^2 as

$$
w^2 = \text{RDERR}^2 + (\text{ERCOF} \cdot \text{RMS}^2)
$$

where RDERR is set by the ERR command, ERCOF is set by the ERC command, and RMS is the root-mean-square travel time residual. RDERR represents the aggregate of all un-modeled timing errors, including estimated reading error in seconds of the arrival time data. ERCOF is just a weighting factor for including the effects of a poor solution (large RMS) in the error calculations. If you want the calculated errors in the hypocenter to reflect only the estimated errors, set ERCOF = 0. This will give errors that include primarily the effects of array geometry.

If you want to include effects of poorly modeled travel times such as uncertainties in the crustal or delay models, then set ERCOF = 1. ERCOF can be set to any positive value or 0.

The covariance matrix is a 4 x 4 symmetric matrix whose diagonal elements are the variances (standard errors squared) of origin time (in sec), and latitude, longitude and depth (all in km). The off-diagonal elements are the covariances between these quantities. This allows, for example, a quantitative estimate of origin time error and the tradeoff between origin time and depth. The error ellipsoid is specified by the 3 x 3 sub-matrix with origin time removed.

Error ellipsoid and vertical and horizontal errors

The error ellipsoid is specified by the 3 x 3 sub-matrix derived by removing origin time from the covariance matrix. The 3 x 3 spatial covariance matrix must be rotated into the principal coordinates of the solution, whose axes are the major axes of the error ellipsoid. The three principal standard errors are calculated by taking square roots of the eigenvalues (diagonal elements in diagonal form) of the 3 x 3 covariance matrix. The earthquake then has a statistical probability of 32% of lying inside an ellipsoid of error whose major axes are given by the three principal standard errors. An error ellipsoid whose major axes are 2.4 times the standard errors calculated by this program has a 95% chance of containing the "true" hypocenter. Hypoinverse also calculates the azimuths and dips of the principal axes of the error ellipsoid.

Note that the calculated error ellipse scales with the estimated error RDERR and the earthquake's RMS. The error ellipse is thus also only an estimate, though it contains all the geometry of the stations and ray paths. A true calculation of the error ellipse would require knowing the uncertainties in every variable, including the crustal model, which is nearly impossible. In addition, the error ellipse is based on the partial derivatives at the hypocenter, and knows nothing about real or modeled discontinuities such as layer boundaries or the Earth's surface.

The vertical error ERZ and horizontal error ERH are simplified errors derived from the lengths and directions of the principal axes of the error ellipsoid. Each of the three principal axes (whose lengths are the standard errors) are projected onto a vertical line through the hypocenter, and the largest value is ERZ. ERH is simply the length of the longest of the principal axes when viewed from above and projected onto a horizontal plane.

Appendix 5: Generating travel-time tables for linear gradient crustal models with program TTGEN

Use of a travel-time table

Hypoinverse reads a travel-time table, which is generated independently of the location process by the program TTGEN. Hypoinverse calculates travel time, travel time derivatives, and emergence angles at the source by interpolation from this table. Three-point (parabolic) interpolation is used. Linear extrapolation is used beyond the table, which is exact for refractions from the underlying halfspace. The table itself is a condensed grid of travel times as a function of distance and depth. Two different grid point spacings are permitted for each of distance and depth, so that travel times for shallow nearby sources may be accurately modeled with close spacing, without wasting computer memory on deep or distant grid points where the travel time curve changes slowly. The user may generate his own travel time table empirically

or with another program (see later for table format) or use the travel-time generating program TTGEN to prepare a table from a given velocity-depth function.

Allowable crustal models input to TTGEN

Crustal models consist of from 2 to 15 points at which the user specifies velocity and depth. Linear velocity gradients are assumed to connect the points. The last point fixes the velocity and depth of the homogeneous half-space underlying the model. The halfspace velocity must be the greatest of any of the velocities specified to insure that rays can be refracted along the top of the halfspace. Rays traveling through layers with a linear velocity gradient trace circular ray paths when viewed from the side. Rays can thus bottom inside a layer with a gradient. Rays in a homogeneous layer trace a circular ray path with infinite radius (a straight line).

The use of linear gradients smoothes out the discontinuities in travel time derivatives which result from homogeneous layer models, and gives a more realistic spread in emergence angles of down-going rays than is possible with modeling rays as refracted from discontinuities. One buried low velocity zone is permitted in the model. This means that velocity may not decrease with depth except for one group of adjacent velocity points. Hypocenters that occur within a low velocity zone may produce a shadow zone at the surface, and rays in this distance range are calculated as if refracted along the layer above the low velocity zone.

TTGEN can handle models with homogeneous layers (zero gradients), but velocity discontinuities (infinite gradients) are not allowed. Velocity gradients should assume reasonable values such as 0.0 or between 0.02 and 8.0 km/sec/km in the interest of numerical stability.

TTGEN operates by shooting rays out from the source and calculating time, distance, and other parameters where (and if) they emerge at the surface. Layers with steep gradients (such as might be used to model a Moho transition) can produce reverse branches in the travel time curve, and such layers should be at least 0.3 km thick to insure that enough rays will bottom in the layer to define the travel time curve and its reverse branch properly. Errors can be introduced in the final travel-time table by under-sampling a too-complicated or irregular velocity model with too few rays.

Using the program TTGEN

TTGEN shoots rays with increasing ray parameter starting with vertically emergent rays, and calculates distance, travel time, and other parameters for each ray (see outputs of TTGEN section), all at depth intervals specified by the user. At each depth, a printed listing of these results is produced, noting any reverse branches or rays lost to a low velocity waveguide. At the same time, a file of distance, travel time, reduced travel time and emergence angle (from nadir) is made for all branches of the travel time curve, which can be plotted later (see figure 8). A separate file is produced for each depth, with the model name forming the base filename and depth as the file extension. The program then produces the final travel-time table by interpolating travel times at regular distance intervals specified by the user. The travel-time curve used for earthquakes is the first arrival from among the various branches including refractions from the halfspace and top of any low velocity zone.

Input to TTGEN in the file TTMOD

All model parameters including depth, distance, and ray intervals at which computations are to be performed are input to TTGEN in the file "ttmod". The program uses reduced travel times for

the table to save space. You specify the inverse of the reducing velocity REDV (in sec/km). The reduced travel time is the absolute travel time minus distance times REDV. The values of reduced travel time read by Hypoinverse are limited to the range 0 to 32 seconds, and the user is responsible for choosing a suitable reducing velocity to stay within these limits. Using a reducing velocity equal to the halfspace velocity is a good choice.

The user specifies the amount by which the independent parameter Q is incremented to calculate distance and time for rays of various ray parameter and emergence angle. Ray parameter P and emergence angle _ are functions of Q as follows:

$$\_ \ = \ 2 \ \bullet \ TAN^{-1}[\ Q/(Z_H \ + \ \_)]$$

$$p \ = \ SIN(\_)/V_H$$

where Z_H and V_H are depth and velocity at the hypocenter, respectively. Q is a better independent parameter than either P or _ since it gives a greater density of rays for deeper penetrations. This also gives the distant travel time points a spacing in distance comparable to nearby points.

The parameter Q is incremented as follows. It takes on the value 0.0, then NQ1 values at increments of DQ1, then NQ2 values at increments of DQ2. The largest value of Q is thus NQ1 • DQ1 + NQ2 • DQ2, and the greatest number of rays (maximum value of NQ1 + NQ2) is 400. Ray calculation stops when down-going rays begin to penetrate the halfspace. Travel times appropriate to a refracted ray are used beyond this point. Values of DQ1 = 0.04, NQ1 = 200, DQ2 = 0.2, and NQ2 = 200 are a good first try, and generally insure that the entire travel time curve can be adequately defined by less than 400 rays.

The grid points in distance and depth at which travel times are calculated for output to the final table are determined by eight parameters similar in concept to the Q parameters described above. Travel times are calculated at depths of 0.0, then NZ1 values at increments of DZ1, then NZ2 values at increments of DZ2. This permits a fine grid spacing for shallow depths and a coarse spacing at greater depths where the travel time curve will be smoother. Similarly, travel times are calculated at distances of 0.0, then ND1 values at increments of DD1, then ND2 values at increments of DD2. Presently the maximum value of NZ1 + NZ2 is 27, and ND1 + ND2 may be as large as 41. These maximum array sizes can easily be increased by increasing the array dimensions and then recompiling.

Velocity model input format (file 'ttmod.')

Start Col.	Len.	Fortran Format	Data
Line 1:			
1	8	A8	Printed output filename.

| 9 | 8 | A8 | Travel time table output filename. The name must end in a period (.). The extension 'crt' will be added to indicate it should be read with the Hypoinverse CRT command. |
| 17 | 6 | F6.1 | REDV, one divided by the reducing velocity used to condense the travel time tables. |

Line 2: Parameters for inerementing the independent parameter Q governing ray spacing (see text).

1	5	F5.2	DQ1
6	5	I5	NQ1
11	5	F5.2	DQ2
16	5	I5	NQ2

Line 3: Parameters for inerementing the grid spacing in depth (see text).

1	5	F5.2	DZ1
6	5	I5	NZ1
11	5	F5.2	DZ2
16	5	I5	NZ2

Line 4: Parameters for incrementing the grid spacing in distance (see text).

1	5	F5.2	DD1
6	5	I5	ND1
11	5	F5.2	DD2
16	5	I5	ND2

Line 5:

| 1 | 20 | A20 | Name of model. The first 3 letters of the 20 are the model code used throughout Hypoinverse and the output files as a model label. I prefer to make the model code the same as the root of the travel-time table output file name. |

Line 6 and later:

| 1 | 5 | F5.2 | Velocity of first point (km/sec). |
| 6 | 5 | F5.2 | Depth of first point (km). This format is repeated for each velocity- depth point of the model, one line per point, up to a total of 15 points. The last point given sets the velocity and depth of the halfspace. |

Here is a sample input file ("ttmod."). The moho is represented by a thin layer with high gradient between 13.7 and 15.5 km depth. The halfspace under the model has an upper mantle velocity of 8.3 km/sec. The reducing velocity is $1/0.13 = 7.69$ km/sec.

```
hg3.PRT hg3.    .130
  .08  100  0.4   90   (48 MAX)
  1.0   18  4.0    9   (40 KM DEPTH MAX)
```

```
   1.0   28   4.0   13   (135 KM DIST MAX)
hg32
   1.9  0.0
   3.0  1.4
   6.2  3.5
   7.2 13.7
   8.3 15.5
```
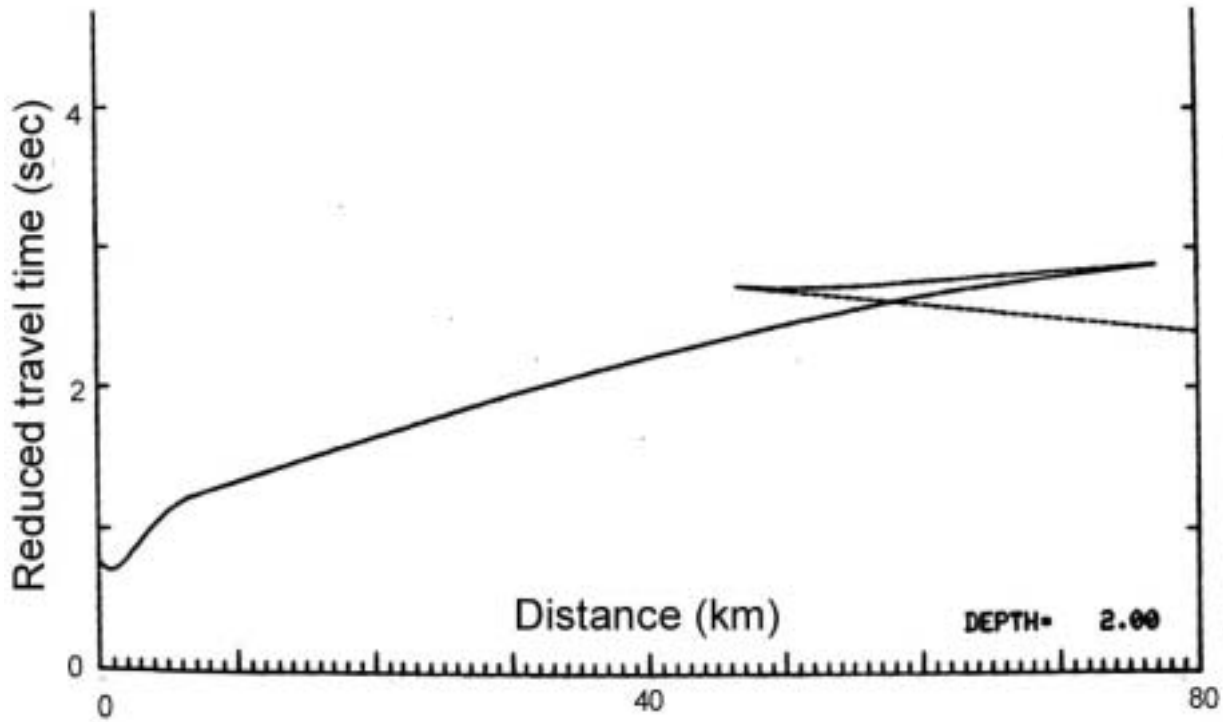

Figure 8. A sample plot of the reduced travel time curve for the above input file for a source at 2 km depth.

The straight branch (dashed) is the Pn branch refracted from the halfspace. The reverse branch is from rays which bottom in the thin layer with a high gradient that represents the moho. Because the final travel time curve is the earliest (lowest) of each of the branches, the reverse branch is not present in the travel-time table and the curve behaves like that of a refracting surface. The thin high-gradient layer also smoothes out the depths of hypocenters near it because there is no sudden discontinuity.

Outputs of TTGEN

At each depth point specified in the grid, a plot file is created of travel times and distance. The actual travel time used in Hypoinverse will be the first arrival from each branch. All reversed branches are indicated. The condensed travel-time table contains all the information necessary to identify itself and be used by Hypoinverse when read with the CRT command. The format of the table is transparent to the user, but is given below for completeness.

The print output of TTGEN contains one table for each depth grid point. One line is printed for each ray calculation until the deepening rays reach the halfspace. The tabulated data is as follows:

117

J	The ray index, an integer 1, 2, 3... Also used to reference rays defining the endpoints of a shadow zone or reversed branches.
Q	The user-defined parameter variable. Equal increments of Q are designed to give a greater density of deeper rays where they are needed to define the travel time curve.
EM.ANG	Emergence angle of ray at the source, measured in degrees from zenith.
P	Ray parameter in sec/km.
DIST	Distance in km at which ray reaches the surface. If DIST = -1, then the ray is trapped in a waveguide and does not reach the surface.
TIME	Travel time in seconds.
REDUCED	Reduced travel time in seconds, given by TTIME - DIST • REDV, where REDV is one divided by the reducing velocity.
L.BOT	The layer in which down-going rays bottom.
Z.BOT	The depth at which down-going rays bottom.
V.BOT	The velocity at which down-going rays bottom.
DDIF	The difference in distance between this and the preceding ray. DDIF is negative on reverse branches.
BR	Branch number. It is incremented by 1 each time a new forward branch is encountered.
AMP	The relative amplitude of the ray at the surface assuming an isotropic source and geometrical spreading. It is just the ratio of the area of a ring on a unit sphere surrounding the source to the corresponding area at which rays emerge at the earth's surface.
AMP*R**2	Amplitude times distance squared. Used to estimate the differences between actual and ideal inverse-square spreading.
REMK	Remark such as RB (reversed branch) or WG (ray in wave guide).

TTGEN also writes a series of files for plotting travel time curves such as figure 8. Each filename is composed of the model name (the "travel time table output filename" input on line 1 of the file ttmod). The file extension part of the name is a 3-digit number indicating the depth of the source in 0.1 km. Thus hg3.050 is a possible name for data for the depth 5.0 km. Each line of the file contains four numbers:

DIST	Distance in km at which ray reaches the surface. If DIST = -1, then the ray is trapped in a waveguide and does not reach the surface.
TIME	Travel time in seconds.
REDUCED	Reduced travel time in seconds, given by TTIME - DIST • REDV, where REDV is one divided by the reducing velocity.
INS.ANG	Incidence angle of ray at the source, measured in degrees from nadir. This is 180 - EM.ANG, the emergence angle output to the print file.

Format of the travel time table generated by TTGEN and used by Hypoinverse

Start
Col.   Len.   Fortran
Col.   Len.   Format   Data

Start Col.	Len.	Fortran Format	Data
Line 1:			
1	20	A20	Model title to appear on output.
21	2	I2	Number of velocity points n specified for the model.
23	8	F8.4	One over the reducing velocity (in sec/km) used to reduce travel times in table.
Line 2:			
1	75	nF5.2	Depths of the n points of model.
Line 3:			
1	75	nF5.2	Velocities of the n points of model.
Line 4:			
1	7	F7.4	DD1
8	3	I3	ND1
11	7	F7.4	DD2
16	3	I3	ND2
Line 5:			
1	7	F7.4	DZ1
8	3	I3	NZ1
11	7	F7.4	DZ2
16	3	I3	NZ2

Travel time Blocks.
One block now follows for each of the NZ1 + NZ2 + 1 depth grid points.

Start Col.	Len.	Fortran Format	Data
Line 1 of block:			
1	10	F10.4	Depth.
11	10	F10.4	Velocity at this depth.
21	10	I10	Distance in units of 0.1 km at which a horizontally emergent ray reaches the surface. This is used to resolve the ambiguity between upgoing and downgoing rays.
Line 2 etc. of block:			
1	90	15I6	Reduced travel times at each of the ND1 + ND2 + 1 distance grid points. The reduced travel times are given as integers in units of .0005 sec, minus 32000. Ths converts an integer range of +/- 32000 into a time range of 0.0 to 32 seconds. This encryption was done because Hypoinverse was originally

> programmed on a 16-bit computer with two-byte integers to save space.

Appendix 6: Programmers' notes

The source code is in fortran77 with a few of the extensions. The code is compatible with Sun computers running the sunos or solaris operating systems, and DEC/Compac VAX or Alpha computers running VMS. System-dependent fortran code, like OPEN statements and system calls, have been put in separate and small subroutines, which can be linked into the appropriate executable. For example, the subroutine openr.f opens files for reading on unix systems, and OPENR.FOR does the same on VMS systems. When both a *.f and *.FOR version are present, the former is for the unix system and the *.FOR is for the VMS system.

REFERENCES

Bakun, W.B., and W. Joiner, 1984, The ML scale in central California, Bull. Seis. Soc. Am, v. 74, p1827.

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. Geological Survey Open-File Report.

Eaton, J. P., 1970 and later, Harmonic magnification of the complete telemetered seismic system, from seismometer to film viewer screen, U.S. Geological Survey Open-File Report, 23 pp.

Eaton, J. P., 1992, Determination of amplitude and duration magnitudes and site residuals from short-period seismographs in Northern California, Bull. Seis. Soc. Am, v. 82 no. 2, pp. 533-579.

Geiger, L., 1912, Probability method for the determination of earthquake epicenters from the arrival time only (translated from Geiger's 1910 German article), Bulletin of St. Louis University, v. 8 no. 1, pp. 56-71.

Lahr, J.C, 1980, HYPOELLIPSE: a computer program for determining local earthquake hypocentral parameters, magnitude and first motion pattern, U.S. Geological Survey Open-File Report 80-59, 59 pp.

Klein, F.W, 1978, Hypocenter location program HYPOINVERSE, U.S. Geological Survey Open-File Report 78-694, 113 pp.

Klein, F.W, 1985, HYPOINVERSE, a program for VAX and Pro-350 computers to solve for earthquake locations and magnitudes, U.S. Geological Survey Open-File Report 85-515.

Klein, F.W, 1989, HYPOINVERSE, a program for VAX computers to solve for earthquake locations and magnitudes, U.S. Geological Survey Open-File Report 89-314, 59 pp.

Lee, W. H. K., and Lahr, J. C., 1972, HYPO71: A computer program for determining hypocenter, magnitude, and first motion pattern of local earthquakes, U.S. Geological Survey Open-File Report.

Lee, W. H. K., R.E. Bennet, and K.L. Meagher, 1972, A method of estimating magnitude of local earthquakes from signal duration, U.S. Geological Survey Open-File Report, 28 pp.

Lamson, C. L., Hanson, R. L., 1974, *Solving Least Squares Problems*, Prentice Hall, 340 pp.

Michaelson, C.A., 1987, Coda duration magnitudes in Central California, U.S.G.S. Open File Report 87-588.

INDEX OF COMMANDS AND SPECIAL NAMES

Page numbers in bold are the complete command explanations in the command dictionary.

The old commands LES, DLY, ST5 and VER are no longer used, but are recognized and produce an error message identifying the replacement command. The p-amplitude magnitude commands PAC, PC1, PC2, PMA and PMC are functional, but not fully implemented or documented.