# Kent Lindquist White Paper

Number 2001-001

# Error Ellipses for AEIC Earthquake Catalogs

*Kent Lindquist*

January 3, 2001

# Error Ellipses for AEIC Earthquake Catalogs

The Alaskan earthquake catalogs have been compiled with multiple earthquake location programs throughout the years, all of which have produced some form of uncertainty information associated with each hypocenter. This report does not attempt a comprehensive survey of all the issues involved. Rather, the intent here is to give an acceptably concrete description of what is currently in our catalogs, bulletins, and web pages. This does not preclude further examination of the error-ellipse issue in the future. An appendix provides detailed information on the steps used to pull our current earthquake catalogs and web-based weekly reports into line with the specification in this report.

For solutions with dbgenloc (August, 1999 and beyond, plus part of July 1999), a covariance matrix is calculated during the inversion for earthquake location. This covariance matrix is then scaled to a 68.3% confidence surface (one standard-deviation) using a chi-squared statistical model. Two values are shown on the web. The SEZ value is the projection of that confidence surface on the vertical axis, scaled with the chi-squared critical value for one-degree-of-freedom at 68.3% confidence. The SEH value is the semi-major axis of the horizontal, surface error-ellipse, scaled with the chi-squared critical value for one degree-of-freedom at 68.3% confidence. Further details on the construction of the covariance matrix by dbgenloc may be taken from the man page on eqerror(3) in the Antelope contributed-code distribution:

> Output covariance matrix of estimated hypocenter. The order of parameters within the matrix is x (east-west), y (north-south), z (vertical), and time. (e.g. C[1][2] = C_yz ) This matrix is scaled according to the set of combined weights that are used in the matrix formed by the equations of condition for the final hypocenter solution. In genloc this is the product of three terms: (1) base weighting by 1/sigma where sigma is the input measurement uncertainty, (2) residual weighting (if turned on), and (3) distance weighting (if turned on). As such this covariance can be turned into a measurement error based estimate directly with a chi-squared formula (see below).

In the generation of AEIC catalogs we have had the residual weighting (the dbgenloc *arrival_residual_weight_method* parameter) turned off and the distance weighting turned on (the dbgenloc *time_distance_weighting* parameter plus weighting factors defined in the velocity models in blocks of name *time_distance_weight_function*).

The discussion in the same man page of the chi-squared error model used is also illuminating:

> Error model to use to compute confidence ellipse. Allowed values are CHI_SQUARE and F_DIST (defined in location.h). The CHI_SQUARE was used, for example, in hypoellipse. It assumes the covariance is a perfect estimate of uncertainty and scales and projects this ellipse assuming the original error estimates used to weight the measurements are exact. It is best thought of as a measure of the precision of the location in relation to measurement errors. F_DIST uses an F-distribution to scale the error ellipse using the rms variable (see below) according to the classic paper by Flinn (1965). The concept is rms scaling is supposed to measure location accuracy including model errors. In practice, however, it is now known this is

not a reliable error model. It is used as an option largely as a way to get results that can be compared to older programs.

Older earthquake solutions in the AEIC catalog (June, 1999 and earlier, plus parts of July, 1999) have been generated with the program Hypoellipse by John Lahr. Similar to the dbgenloc method described above, Hypoellipse generates the same SEH and SEZ error estimates at 68.3% confidence level for one-degree of freedom with a Chi-squared statistical model. This Hypoellipse reporting method for hypocentral confidence intervals in fact formed the basis of AEIC's current monthly-catalog and web-based catalog format. The dbgenloc output was retrofitted to match. Further discussion may be warranted in the future about whether or not we continue reporting error statistics in this way. In any case, there are several available sources of documentation for the details of error ellipse creation by Hypoellipse:

Lahr, J.C. (1996). Hypoellipse/Version 3.0: A computer program for determining local earthquake hypocentral parameters, magnitude, and first-motion pattern. US Geological Survey Open-file report 96-xxx.

Lahr, J.C. (1984). Description of the weighted regression and quality estimation used in the earthquake location program Hypoellipse. *U.S. Geological Survey Open-file Report* **84-766**.

Boyd, T.M. and J.A. Snoke (1984). Error Estimates in some commonly used earthquake location programs. *Earthquake Notes* **55**, Number 2, pp. 3-6.

# Appendix A

We have endeavored to fix the error-ellipse difficulties for hypocenters in AEIC catalogs. This has taken place in several broad strokes: first to correctly project the covariance ellipse in the CSS3.0 origerr tables; then to correct a bug in the pickfile translation program along with its consequences. Partway into the task (after an initial, mistaken attempt to correct a number of scratch databases used for the generation of weekly reports), I decided this process was involved enough to merit extensive note-taking. The ensuing stages of work are documented here.

Beginning on 11/22/00

1) First, I already made a program called dbproject_covariance, and used it to change some of the weekly-report databases in Dan McNamara's home directory. Unfortunately this first version of dbproject_covariance was not correct, therefore we have to reverse those changes. I also used those modified databases to regenerate our web-page weekly reports, however in this step I am just going to back out the changes made to the weekly databases in Mcnamara's home directory. From my file /home/kent/work/fix_ellipse/Weekly_database_list (compiled by Trilby Cox though I made the pathnames absolute and changed a couple typos), we see the databases that were affected by my initial incorrect projection:

/home/mcnamara/AEIC/Weekly_Report/1999/July/July19-25/jul1925db
/home/mcnamara/AEIC/Weekly_Report/1999/July/July26_Aug1/july26db
/home/mcnamara/AEIC/Weekly_Report/1999/August/Aug9_15/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/August/Aug16_Aug22/weekly_db
/home/mcnamara/AEIC/Weekly_Report/1999/August/Aug23_Aug29/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/August/Aug30_Sept05/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/September/Sept6_Sept12/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/September/Sept13_Sept19/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/September/Sept20_Sept26/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/September/Sept27_Oct3/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/October/Oct4_Oct10/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/October/Oct11_Oct17/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/October/Oct18_Oct24/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/October/Oct25_Oct31/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/November/Nov1_Nov7/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/November/Nov8_Nov14/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/November/Nov15_Nov21/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/November/Nov22_Nov28/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/November/Nov29_Dec5/weeklydb
/home/mcnamara/AEIC/Weekly_Report/1999/December/Dec6_Dec12/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Dec27_Jan2/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Jan3_Jan9/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Jan10_Jan16/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Jan17_Jan23/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Jan24_Jan30/weeklydb

/home/mcnamara/AEIC/Weekly_Report/Jan31-Feb6/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Feb7_Feb13/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Feb14_Feb20/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Feb21_Feb27/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Feb28_Mar5/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Mar6_Mar12/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Mar13_19/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Mar20_Mar26/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Mar27_Apr2/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Apr3_Apr9/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Apr10_Apr16/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Apr17_Apr23/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Apr24_Apr30/weeklydb
/home/mcnamara/AEIC/Weekly_Report/May1-May7/weeklydb
/home/mcnamara/AEIC/Weekly_Report/May8_May14/weeklydb
/home/mcnamara/AEIC/Weekly_Report/May15_May21/weeklydb
/home/mcnamara/AEIC/Weekly_Report/May22_May28/weeklydb
/home/mcnamara/AEIC/Weekly_Report/May29_June4/weeklydb
/home/mcnamara/AEIC/Weekly_Report/June5_June11/weeklydb
/home/mcnamara/AEIC/Weekly_Report/June12_June18/weeklydb
/home/mcnamara/AEIC/Weekly_Report/June19_25/weeklydb
/home/mcnamara/AEIC/Weekly_Report/June26_July2/weeklydb
/home/mcnamara/AEIC/Weekly_Report/July3_July9/weeklydb
/home/mcnamara/AEIC/Weekly_Report/July10_July16/weeklydb
/home/mcnamara/AEIC/Weekly_Report/July17_July23/weeklydb
/home/mcnamara/AEIC/Weekly_Report/July24_July30/weeklydb
/home/mcnamara/AEIC/Weekly_Report/July31_Aug6/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Aug7_Aug13/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Aug14_Aug20/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Aug21_Aug27/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Aug28_Sept3/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Sept4_Sept10/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Sept11_sept17/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Sept18_Sept24/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Sept25_Oct1/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Oct02_Oct08/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Oct9_Oct15/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Oct16_Oct22/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Oct23_Oct29/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Oct30_Nov5/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov6_Nov12/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov13_Nov19/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov20_Nov26/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov27_Dec3/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Dec4_Dec10/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Dec11_Dec17/weeklydb

/home/mcnamara/AEIC/Weekly_Report/Dec18_Dec24/weeklydb

The way that dbproject_covariance was run, it projected the covariance matrices only for the genloc solutions. Thus it is only those which we need to back out. The dbproject_covariance program ignored any rows that had any elements of the covariance matrix null in the database. Also, when I ran dbproject_covariance I chose a confidence level of 0.683. Pick2db solutions have null covariance matrices in these weekly databases. dblocsat solutions have non-null covariance matrices and non-null smajax/sminax/strike/sdepth/stime, but the confidence level is always at 0.9. I tested this with

```
nordic% foreach db (‘cat Weekly_database_list‘)
foreach? dbjoin $db.origerr origin | dbsubset - ’algorithm =~ /locsat.*/’ | dbselect -  conf | uniq
foreach? end
```

Only the recently and incorrectly-projected dbgenloc solutions in these databases have origerr.sxx non-null as well as origerr.conf != 0.9. Therefore we can use those two criteria to null-out the incorrectly projected fields origerr.smajax, sminax, strike, sdepth, stime, and conf. After testing the procedure on a copy of the first database in the above list (convenient because it has examples of all three types of origerr entries, those from pick2db, dblocsat, and dbgenloc), we accomplish the nulling out of the incorrect projections as follows:

```
nordic% foreach db ( ‘cat Weekly_database_list‘ )
foreach? echo $db >>! notes_nullout_projections
foreach? dbset -v $db.origerr smajax ’sxx != NULL && conf != 0.9’ -1 >>&!
notes_nullout_projections
foreach? dbset -v $db.origerr sminax ’sxx != NULL && conf != 0.9’ -1 >>&!
notes_nullout_projections
foreach? dbset -v $db.origerr strike ’sxx != NULL && conf != 0.9’ -1 >>&!
notes_nullout_projections
foreach? dbset -v $db.origerr sdepth ’sxx != NULL && conf != 0.9’ -1 >>&!
notes_nullout_projections
foreach? dbset -v $db.origerr stime ’sxx != NULL && conf != 0.9’ -1 >>&!
notes_nullout_projections
foreach? dbset -v $db.origerr conf ’sxx != NULL && conf != 0.9’ 0 >>&!
notes_nullout_projections
foreach? end
```

That notes file is currently in /home/kent/work/fix_ellipse/notes_nullout_projections. It’s 34,317 lines and takes 1.7 MB, to give an idea of the scale.

2) Now I’m going to do something extremely bold. I have a fair degree of confidence that I will succeed. The alternative is to leave the whole confidence-ellipsoid mess unresolved until after AGU. I am going to import and install Gary’s new dbgenloc that projects confidence ellipses.

First, in contrib/lib/location/libgenloc the following files changed:
Makefile eqerror.c locate.c location.h

Those are currently at the following revision levels:

Makefile 1.10
eqerror.c 1.5
locate.c 1.12
location.h 1.7

Thus the recovery strategy in this directory, should it be necessary, will be

cvs update -r 1.9 Makefile
cvs update -r 1.4 eqerror.c
cvs update -r 1.11 locate.c
cvs update -r 1.6 location.h

Don't forget that CVS update tags are sticky!!!! [i.e. their state can persist through updates--see CVS documentation]. If you recover this way you will need to clean up down the road!!

Second, in contrib/bin/location/dbgenloc the following files changed:
Makefile dbgenloc.h dbgenloc.pf run_location.c save_results.c write_log.c

These are currently at the following revision levels:

Makefile 1.8
dbgenloc.h 1.3
dbgenloc.pf 1.3
run_location.c 1.5
save_results.c 1.10
write_log.c 1.3

Thus the recovery strategy in this directory, should it be necessary, will be

cvs update -r 1.7 Makefile
cvs update -r 1.2 dbgenloc.h
cvs update -r 1.2 dbgenloc.pf
cvs update -r 1.4 run_location.c
cvs update -r 1.9 save_results.c
cvs update -r 1.2 write_log.c

Once again if you take this route after all is cleaned up we will have to cvs update to the tag HEAD to get back in sync with the contrib area in Indiana. Let's hope this does not become necessary.

Given that, I have made both of these directories and installed them (preceded by make Include).

Gary added the parameters confidence and ellipse_type to dbgenloc.pf, but not dbgenloc_default.pf in the contrib area. I believe this is an oversight, which I've pointed out. [N.B. on 12/29/00 I added this in to contrib myself]. We will work around this because we maintain and install our own local-mod version of dbgenloc_default.pf via the /usr/local/aeic package. Thus in this area I added

confidence 0.683
ellipse_type chi_square

to our dbgenloc_default.pf and installed the change. In order to support this I also had to design a couple changes to /opt/antelope/4.2u/data/tcl/library/dbloc/dbgenloc.tcl, which I made and sent to Quinlan.

Unfortunately the dbgenloc error-ellipse projection looks problematic, so I will have to back out of the software update. I am going to leave my dbgenloc_default.pf local mod in place since it will probably need to be there anyway and is harmless and forward-looking.

Continuing on 12/29/00.

2 (continued)) Gary has apparently fixed the bug that caused the negative sdepth and stime in the genloc projection routines. I was about to import and re-test this here, however the CVS disk crashed in Indiana, making that a bridge-burning step (i.e. if something goes wrong we will not be locating earthquakes until CVS checkout works again). Now CVS checkout works again, I have imported the new genloc, and I have verified that we have the bug-fixed copy and that if something goes wrong I can use the same recovery commands as I have listed above.

As it is, after testing I believe the only thing wrong with the new genloc error projection is that the phase of the strike angle needs to be wrapped to account for a mismatch between the atan2 principle value range and the range of the origerr.strike field in CSS3.0. I fixed this, then relocated a day's worth of events (12/28/00) to look for reasonableness of origerr results.

I am going to leave the new dbgenloc in place. According to the tracking database, everything through and including the day 12/26/00 has been processed. Thus all of this has been done with the old dbgenloc. Three days, 12/27-29/00, are currently checked out, thus they will get processed with the new dbgenloc. All days 12/30/00 and onwards will get processed completely with the new dbgenloc unless I have to back something out that I've done so far.

3) I recast dbproject_covariance to use Gary's project_covariance routine; then I ran it against Trilby's original solutions for 12/28/00 and compared the error ellipses in my relocations with the new genloc to the projections of the error ellipses from the old genloc. At least these agree, which is a reasonable internal consistency check. At Roger's suggestion I also ran all these locations with locsat and the iasp91 model. Some didn't converge, however for those that did the strikes of the error ellipses agree moderately well to very well (within a few degrees in many cases. Specifically, for the 8 earthquakes the strikes are within, respectively, [0.6, 0.8, 1.86, 0.07, 0.29, 1.97, 0.12, 10.36] degrees). The one subtlety is that locsat apparently constrains the strike of the error ellipse to be 0 to 180 degrees, whereas the current genloc library allows it to go fully 0 to 360 degrees. Since there is nothing like a "down-dip" direction here, the meaningful information in the strike axis wraps every 180 degrees instead of 360, so we could conceivably fix dbgenloc to standardize this. In any case that will be an easy conversion to do later so for now I will leave it as it is.

As for the sizes of the error ellipses, the genloc and the locsat versions are of the same order of magnitude, with the genloc solutions usually a bit smaller--too much so to be explained by the difference in confidence surface, i.e. choice of critical chi-squared value. The smajax/sminax (2-degrees of freedom) values should be divided by

$$\sqrt{\frac{\chi^2(0.9conf)}{\chi^2(0.683conf)}} = \sqrt{\frac{4.60517}{2.297707}} = 1.24$$

in order to take a 90% confidence value to a 68.3% confidence value, quoting critical chi-squared values out of revision 1.7 of Gary's contrib/lib/location/libgenloc/eqerror.c. Similarly, for the one-degree of freedom quantities of sdepth and stime, to get from 90% confidence to 68.3% confidence the values should be divided by

$$\sqrt{\frac{\chi^2(0.9conf)}{\chi^2(0.683conf)}} = \sqrt{\frac{2.705543}{1.001284}} = 1.64$$

Specifically, for the 8 events for which I have a locsat solution, after adjusting smajax from locsat down to a 68.3% confidence level, the locsat values for smajax are still bigger than the dbgenloc values by factors of [1.56, 1.46, 3.94, 1.41, 3.19, 1.54, 1.24, 1.47] respectively for the 8 earthquakes. I am not going to let this level of disagreement stop me from plowing on through to completed web pages with error ellipses.

4) The next step is a preparatory detail, as well as perhaps another small test before the final plunge. I am going to sneak into ed, trilby, and martin's home directories for the 12/27.28.29/00 days respectively and project their half-finished processing databases, and tmp/trial databases, so when they check them in all the origerr rows will be correct. This projection seems to have gone smoothly, based on quick visual inspection of the origerr tables.

5) Now I will run dbproject_covariance on the weekly-report databases. Since the time of my last efforts, nine more weekly reports have been made. I have taken the file

/home/kent/work/fix_ellipse/Weekly_database_list

and added to it the following databases:

/home/mcnamara/AEIC/Weekly_Report/Oct23_Oct29/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Oct30_Nov5/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov6_Nov12/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov13_Nov19/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov20_Nov26/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Nov27_Dec3/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Dec4_Dec10/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Dec11_Dec17/weeklydb
/home/mcnamara/AEIC/Weekly_Report/Dec18_Dec24/weeklydb

Now to project the entire set, we run

```
nordic% foreach db ( ‘cat Weekly_database_list‘ )
foreach? echo $db >>! notes_project_weekly_dbs
foreach? dbproject_covariance -v $db
foreach? end
```

Very good. The only thing I forgot was to redirect the actual output of dbproject_covariance onto the notes file, so said notes file is essentially useless. This is not a reversible situation without backup tapes. However, we have enough trust in the programs so far that everything is probably OK.

6) Next we will run dbproject_covariance on all the databases in the official directories for processed events [the weekly_report databases above are merely convenient intermediate databases that are not for archiving or continued processing--we are dealing with them just because it’s an easy way to regenerate weekly-report web pages]. The official directories for processed events are in /Seis/processing/analyzed, containing databases from July 20, 1999 (actually also including a month in June, 1999) up to the present, i.e. through 12/26,00, everything that has been checked in. This encompasses every dbgenloc solution in our processing and catalog production line.

```
nordic% foreach db ( ‘ls /Seis/processing/analyzed/*/*.origerr | sed -e ’s/.origerr//’‘ )
foreach? echo $db >>! notes_project_analyzed_dbs
foreach? dbproject_covariance -v $db >>! notes_project_analyzed_dbs
foreach? end
```

The resulting notes file is a little under 21,000 lines. Unfortunately I did not have write permission on all the files. Try again, appending a “2” to the name of the notes file. This time it looks like the projection worked.

7) Next we need to fix the error in project_ellipse, which was called by pick2db to take the hypoellipse 3D confidence ellipsoid and project it on the surface. Pick2db also labelled the confidence as 0.68, rather than 0.683. Therefore we will simultaneously change this pick2db hard-wire, using the conf value as a flag to identify which databases have not been fixed. I know this has the flavor of a hack, however it really seems to be the only projecting detail to hang onto. The current project_ellipse.f in /usr/local/aeic/4.2u/src/bin/utility/project_ellipse has, on line 145-146, the step

```
c------ compute maximum in z.
    axz = .5338 /sqrt(v(3,3)-v(1,3)**2/v(1,1)
```

which is the projection of the hypoellipse confidence ellipsoid onto the vertical axis. The axz variable is assigned to the CSS sdepth field elsewhere in the program. This computation is correct:

$$0.5338 = \frac{1}{1.87} = \frac{1}{\sqrt{3.51}} = \sqrt{\frac{\chi^2(1\dot{d}\dot{o}\dot{f})}{\chi^2(3\dot{d}\dot{o}\dot{f})}}$$

for 68.3% confidence level. This is correct because we are scaling from three degrees of freedom for the 3D ellipsoid, down to 1 d.o.f. for the uncertainty bar sdepth on the vertical axis.

Unfortunately, just above this line in project_ellipse.f we have (lines 140-141):

```
ax2 = .5338 /sqrt(ael(1))
ax1 = .5338 /sqrt(ael(3))
```

These are the values that are assigned to the CSS smajax and sminax fields elsewhere in the project_ellipse program. As you can see, the same scale factor is used, going from three d.o.f. to one d.o.f. In fact, what we need instead is the scaling from three to two d.o.f. since smajax and sminax are for a 2-D error ellipse on the surface. Thus the correct value for the horizontal-ellipse calculation in these two lines should be not 0.5338 but rather

$$\sqrt{\frac{\chi^2(2\,dof)}{\chi^2(3\,dof)}} = \sqrt{\frac{2.30}{3.51}} = 0.8095$$

which is larger by a factor of 1.5166. I made this change to project_ellipse.f, and at the same time changed the hard-wire in pick2db from conf=0.68 to conf=0.683. The latter is what pick2db always meant, anyway (double-checked in 11/2000 with John Lahr, as well as the above-mentioned source-code change), so there is now a way to tell if databases generated from pick2db have been corrected for this source-code projection error.

8) Now we need to apply the same fix to all the databases converted to CSS format by pick2db. To do this I wrote a script called correct_pick2db_projection. If the confidence is 0.680 in the origerr table, this script multiplies smajax and sminax by 1.5166 and changes the origerr.conf to 0.683. It complains if the table is not writable. Since the script is short and I do not want it to get lost, I will quote it here:

```
nordic% cat correct_pick2db_projection
: # use perl
eval 'exec perl -S $0 "$@"'
if 0;

use lib "$ENV{ANTELOPE}/data/perl" ;

require "getopts.pl" ;

if ( ! &Getopts('v') || @ARGV != 1 ) {
        die ( "Usage: $0 [-v] database\n" ) ;
} else {
        $dbname = pop( @ARGV );
}
```

```
use Datascope ;

$adj = 1.5166;
# sqrt of chi-square crit for 68.3% confidence,
# 2 degrees of freedom

@db = dbopen( $dbname, "r+" );

@db = dblookup( @db, "", "origerr", "", "" );

if( ! dbquery( @db, "dbTABLE_IS_WRITEABLE" ) ) {
        die( "$dbname.origerr is not writable. Bye!\n" );
}

$nrecs = dbquery( @db, "dbRECORD_COUNT" );

for( $db[3] = 0; $db[3] < $nrecs; $db[3]++ ) {
        ( $orid, $smajax, $sminax, $conf ) =
          dbgetv( @db, "orid", "smajax", "sminax", "conf" );
        next if( $conf != "0.680" );

        $smajax_new = $smajax;
        if( $smajax != -1. ) { $smajax_new *= $adj; }

        $sminax_new = $sminax;
        if( $sminax != -1. ) { $sminax_new *= $adj; }

        if( $opt_v ) {
                print "smajax $smajax -> $smajax_new" .
                     "sminax $sminax -> $sminax_new" .
                     "conf $conf -> 0.683(orid $orid)\n";
        }
        dbputv( @db, "smajax", $smajax_new,
                     "sminax", $sminax_new,
                     "conf", 0.683 );
}
```

Applying this to the weekly-report databases:

```
nordic% foreach db ( ‘cat Weekly_database_list‘ )
foreach? echo $db >>! notes_weekly_fixproj
foreach? correct_pick2db_projection -v $db >>! notes_weekly_fixproj
foreach? end
```

Similarly, for the analyzed events databases:

```
nordic% foreach db ( `ls /Seis/processing/analyzed/*/*.origerr | sed -e 's/.origerr//'` )
foreach? echo $db >>! notes_analyzed_fixproj
foreach? correct_pick2db_projection -v $db >>! notes_analyzed_fixproj
foreach? end
```

Now, finally, we have to apply the correction to the databases in /Seis/catalogs. First make a backup copy. Backup only the origerr tables since the whole databases take up too much space. Make /Seis/space/catalog_staging/aeic/origerr_backup. In /Seis/catalogs/aeic run

```
cp /Seis/catalogs/aeic/*/*.origerr /Seis/space/catalog_staging/aeic/origerr_backup
```

After verifying (i.e. granting) write permissions I applied the correction to the catalogs (7/88 through 5/99) with

```
nordic% foreach db ( `ls /Seis/catalogs/aeic/*/*.origerr | sed -e 's/.origerr//'` )
foreach? echo $db >>! notes_catalog_fixproj
foreach? correct_pick2db_projection -v $db >>! notes_catalog_fixproj
foreach? end
```

I spot checked a couple of these against their backup copies; ran dbcheck on all the catalog databases again (they passed); and removed write permission.

9) Next we need to fix McNamara's program db2weekly.c so it correctly computes seh and sez from smajax and sdepth. The seh and sez are one-dimensional error values at 68.3% confidence. Dan has the following:

```
/* compute SEH and SEZ
        SEH = smajax*(chi square value)
        SEZ = sdepth*(chi square value)
    chi square = 1.87
    see HYPOELLIPSE by J.C. Lahr for details
*/
followed by
        if( smajax != -1 ) {
            seh = smajax * chi;
and
        if( sdepth != -1 ) {
            sez = sdepth * chi;
```

(actually the null-value protection was mine). 1.87 is the square-root of 3.51, which is the chi-squared value for three degrees of freedom at 68.3% confidence. Dan's formula was taken from the caption of Figure 3.1 on page 3.2 of USGS open-file report 96-xxx on Hypoellipse/Version 3.0, according to Dan [Dan also did some comparison tests and apparently got reasonable num-

bers, however we need to touch up the mathematical reasoning a bit here]. Lahr's figure 3.1 caption states

> Error ellipsoid relationships SEH=MAXH/1.87, SEZ=MAXZ/1.87

However, MAXH and MAXZ are not the same as smajax and sdepth. SEZ is the projection of the 68.3% confidence ellipsoid on the vertical axis, scaled to one degree of freedom. origerr.sdepth is the projection of the confidence ellipsoid of level origerr.conf on the vertical axis, scaled to one degree of freedom. SEH is the semi-major axis of the projection of the 68.3% confidence ellipsoid on the surface, scaled to one degree of freedom. smajax is the semi-major axis of the projection of the confidence ellipsoid of level origerr.conf on the surface, scaled to two degrees of freedom. Thus we need to look at scaling these two quantities. We will implement db2weekly to handle only two options, conf=0.683 and conf=0.9, and set the error values to blanks in other cases.

First, scaling sdepth to SEZ. The chi-squared value for one-degree of freedom at 68.3% confidence is essentially one. Thus if conf=0.683, sez = sdepth. If conf=0.9 then to get SEZ we have to multiply sdepth by

$$\sqrt{\frac{\chi^2(1\dot{d}\dot{o}\dot{f}, conf\,0.683)}{\chi^2(1\dot{d}\dot{o}\dot{f}, conf\,0.9)}} = \sqrt{\frac{1}{2.7055}} = 0.6080$$

in order to get SEZ.

Second, scaling smajax to SEH. For origerr.conf = 0.683, we need to scale smajax from two degrees of freedom to one. In other words, to get SEH we need to multiply smajax by

$$\sqrt{\frac{\chi^2(1\dot{d}\dot{o}\dot{f}, conf\,0.683)}{\chi^2(2\dot{d}\dot{o}\dot{f}, conf\,0.683)}} = \sqrt{\frac{1}{2.30}} = 0.6594$$

For origerr.conf = 0.9, we need to scale smajax from two degrees of freedom at 90% confidence to one degree of freedom at 68.3% confidence. In other words, to get SEH we need to multiply smajax by

$$\sqrt{\frac{\chi^2(1\dot{d}\dot{o}\dot{f}, conf\,0.683)}{\chi^2(2\dot{d}\dot{o}\dot{f}, conf\,0.9)}} = \sqrt{\frac{1}{4.6052}} = 0.4660$$

After tweaking db2weekly so it brings in origerr.conf, the relevant code section becomes

```
if( smajax != -1 && conf == 0.683 ) {
        /* sqrt of ratio of chisq for one d.o.f.
          at 68.3% conf (1) over that for two d.o.f.
          at 68.3% conf (2.30) */
        seh = smajax * 0.6594;
        sprintf( seh_string, "%5.2f", seh );
} else if( smajax != -1 && conf == 0.9 ) {
        /* sqrt of ratio of chisq for one d.o.f.
          at 68.3% conf (1) over that for two d.o.f.
```

```
            at 90% conf (4.6052) */
        seh = smajax * 0.4660;
        sprintf( seh_string, "%5.2f", seh );
    } else {
        seh = 0;
        sprintf( seh_string, "" );
    }

    if( sdepth != -1 && conf == 0.683 ) {
        sez = sdepth;
        sprintf( sez_string, "%5.2f", sez );
    } else if( sdepth != -1 && conf == 0.9 ) {
        /* sqrt of ratio of chisq for one d.o.f.
            at 68.3% conf (1) over that for one d.o.f.
            at 90% conf (2.7055) */
        sez = sdepth * 0.6080;
        sprintf( sez_string, "%5.2f", sez );
    } else {
        sez = 0;
        sprintf( sez_string, "" );
    }
```

Actually I compiled two versions of db2weekly, one called db2weekly_nomap that can be used without human interaction to create the files for the archive of weekly reports on the web. To regenerate the web pages, use

```
nordic% abspath fixstuff
/home/kent/work/fix_ellipse/fixstuff
nordic% cat /home/kent/work/fix_ellipse/fixstuff
#!/usr/bin/perl
open(W,"Weekly_database_list" );
while(<W>) {
    chop;
    $dir=‘dirname $_‘;
    chomp( $dir );
    $dbname = ‘basename $_‘;
    chomp( $dbname );
    print $dbname, "",$dir,"\n";

    chdir( $dir );

    system( "/usr/local/seis_apps/db2weekly/db2weekly_nomap $dbname 3.5" );
    system( "/home/trilby/bin/fix_updateweb" );
}
nordic%
```

The /home/trilby/bin/fix_updateweb is a stripped-down version of /usr/tools/scripts/updateweb that just puts in the archive weekly listings. Thus the current weekly report is not up to date with the error ellipses on the main link for current weekly report, however the archived version is correct and when the current weekly report gets replaced next week, everything will be consistent. [N.B. 1/2/01: I ran the full updateweb script in /usr/tools/scripts, so the entire web page should now be up to date].

K. Lindquist 5/18/01