Stress From Slip Data The Michael Way

Documentation as of July 1986
Minor changes made March 1988

This represents a documentation for Andrew Michael's slickenside analysis package as of March 1988. This version includes the bootstrap statistics, with the ability to flip focal mechanisms from one plane to another randomly. For each program I will give the compile command and the run command. The character "%" is used as the prompt. All of this is designed and documented for UNIX Berkeley Standard Distribution 4.3 as implemented on an Integrated Solutions computer. Also included are sample input and output files in the various appendices. The tape is a tar tape that can be read on any UNIX system. The file CONTENTS lists the contents of the tape. The references for this package are:
Michael, A.J. (1984), Determination of stress from slip data: faults and folds, JGR, v. 89, 11,517-11526.
This paper covers the linear stress inversion technique used in the programs slick and slfast.
Michael, A.J. (1987a), The use of focal mechanisms to determine stress: a control study, JGR, v. 92, 357-368.
This paper covers the bootstrap statistics and the problems associated with using focal mechanisms instead of geological field data.
Michael, A.J. (1987b), Stress rotation during the Coalinga aftershock sequence, JGR, v. 92, 7963-7979.
This paper covers the addition of the plane flipping ability to the bootstrap statistics in order to account for uncertainties in the picking of the fault plane from the two nodal planes.

Data File Format

All of the programs start with the same style data file. It has the following format for a data set with n faults.

```
line 1: Comment line
line 2: data for fault one
line 3: data for fault two
 .
 .
 .
line n+1: data for fault n
```

Each data line consists of three numbers separated by blank space. The first number on each line is the dip direction of the fault plane in degrees East of North. The second number on each line is the dip of the fault plane. The third number on each line is the rake of the fault, such that 0 is left lateral motion, 90 is thrust motion, 180 is right lateral motion, and 270 is normal faulting. Of course negative rakes are allowed as are oblique faulting. A sample data file with 9 faults is shown here:

```
Some data from Dixie Valley, NV
100 45 -112
60 45 -104
60 45 -121
115 47 -108
147 47 -85
165 55 -72
135 50 -95
137 48 -92
140 43 -83
```

The whole file is in the file dixie, and is shown in appendix A.

slick

slick is the basic program in the package, it finds the best stress tensor and some information about the misfit based on the technique of Michael (1984).

```
To compile: % make -f makeslick
To run: % slick input_file
Output files made: input_file.oput
e.g. : % slick dixie
   makes dixie.oput
```

See appendix B for an annotated copy of dixie.oput. Note that slick will erase the existing *.oput file and make a new one.

slfast

slfast is a pared down version of slick, it gives only information about the best result and is for use in the bootstrap analysis.

```
To compile: % make -f makeslfast
To run : % slfast input_file
Output files made: input_file.slboot
e.g.: % slfast dixie
   makes dixie.slboot
```

See appendix C for an annotated copy of dixie.slboot. Note that slfast does not erase an existing *.slboot file, instead it appends to it.

bootslickw

bootslickw controls the process of bootstrap resampling a data set many times. It uses slfast to analyze each resampling.

```
To compile: % make -f makebtslw
To run: % bootslickw input_file n w
  where n is the number of times to resample the data
  and   w is the fraction of the times to flip the focal mechanisms
Output files made: Xtemp  Xtemp.slboot
e.g.: % bootslickw dixie 2000 0.1
   will bootstrap resample the dixie data set 2000 times,
   flipping the selected fault and slip direction 10% of the time,
   and run slfast on each resampling.  The output will be in Xtemp.slboot
   while Xtemp will hold the data for the last resampling.
```

Since these are fixed names you can only have one bootslickw running in a directory at a time. Remember that slfast (and hence bootslickw) does not remove the old output file. This is good and bad. To find confidence regions you want a file where the first two lines are the output of slfast on the true data, and the rest are the output of slfast on resampled data. To do this I usually do the following:

```
% rm Xtemp Xtemp.slboot
% cp dixie Xtemp
% slfast Xtemp
% bootslickw dixie 2000
% mv Xtemp.slboot dixie.slboot
```

This leaves the file dixie.slboot with 4002 lines. The first two are the slfast of the true data, and the rest are 2000 slfasts of resampled data sets. An example output file with only 10 resamplings is in Appendix D. Remember it will be hard to duplicate Appendix D because bootslickw seeds the random number generator off the clock. It is important to use a pseudo-random seed, so that you can run the program twice and compare results on a statistical basis and not a deterministic one. The pseudo-random seed also allows you to run bootslickw twice and combine the results, if the same seed is always used this is not valid since the program will always take the same path. The random number generator is designed after the one HP published in the HP-25C program manual. I can't stand the UNIX one, if you can you should probably use it. Just get it to output a uniformly random number on the space 0-1. bootslickw can take a lot of time, depending on the number of resamplings (and your definition of a lot of time). 2000 resamplings takes about 25 CPU minutes on a 750 so you may want to run it in background. (This time is now considerable shorter due to better programming, but you should still run it in background. -AJM, 2/3/88)

plotboots and plotbootso

This program finds the confidence regions given a *.slboot file as described above, that is it must have the slfast result for the correct data at the top of the file.

> To compile: % make -f makeplbts
> To run: % plotboots *.slboot output_file confidence level
> Output files: as named in command line
> e.g. % plotboots dixie.slboot dixie.bplots 95
> will make an output file with the 95% confidence regions for the
> dixie data, based on the resamplings in dixie.slboot.

dixie.bplots will be a onnet input file. onnet is a stereonet plotting program that is explained in it's own documentation which is enclosed. Even if you don't have onnet the output can be used. dixie.bplots is shown in annotated form in appendix E. Also shown in appendix E is the plot that onnet will make from dixie.bplots. Remember that bootstrap resampling will get a better approximation of the confidence region depending on the number of resamplings done. For 80% confidence regions 500 trials seems to be adequate, for 95% confidence regions I have been using 2000 trials. There is also a program called plotbootso with makefile makeplbtso. They are almost identical except for the size of the plot. plotbootso plots every point within the prescribed confidence region, most of these are close to the best result and plot on top of each other. This increases the plotting time and file sizes. Plotboots uses a thinning algorithm to only plot points that are farther than 2 degrees apart and prefers to plot the outside points. The result is a plot that is much quicker to draw and shows the outside of the interval very well. The difference in time can be over a factor of 100. In the documentation is an example of plotboots versus plotbootso.

plata

plata makes data plots.

> To compile: % make -f makeplata
> To run: % plata input_file
> Output files: input_file.plodc
> e.g.: % plata dixie
> makes an onnet input file called dixie.plodc

dixie.plodc and the plot onnet will make are shown in appendix F.

switcher

switcher takes a data file and makes a new one with the other nodal plane as the fault plane.

> To compile: % make -f makeswitch
> To run: % switcher < data_file > new_data_file
> Output files: outputs to standard out.
> e.g. % switcher < dixie > dixies

dixies will be the new data file and is shown in Appendix G.

bothplanes

bothplanes takes a data file and makes a new one with both possible fault planes included.

To compile: % make -f makeboth
To run: % bothplanes < data_file > new_data_file
Output files: outputs to standard out.
e.g.: % bothplanes < dixie > dixieb

dixieb is shown in Appendix H, the format is as follows:

line 1: Comment line
line 2: Original first datum
line 3: Other plane from first datum
line 4: Original second datum
line 5: Other plane from second datum
.
.
.

bootboth

WARNING: I find it difficult to attach any significance to the confidence levels found when using both possible fault planes. I include this program only so you can play with it. If you decide to publish or otherwise distribute any of its results please mention my hesitancy. This issue is discussed in "The determination of stress from focal mechanisms: a control study."

bootboth finds the confidence limits for the analysis when both nodal planes are used as fault planes. It works just like bootslickw except that the data are treated as pairs of faults. To work it must have a data file that has the two possible fault planes for a focal mechanism next to each other in the file just as bothplanes does it.

To compile: % make -f makebtbo
To run: % bootboth input_file n
where n is the number of times to resample the data
Output files: Xtemp Xtemp.slboot
e.g.: % bootboth dixieb 2000
will bootstrap resample the dixieb data set 2000 times
(as pairs) and run slfast on each resampling.

Use plotboots on the bootboth output in the same way you would on bootslickw output. Since bootboth uses slfast to do the real work it also uses Xtemp and Xtemp.slboot. This can be a little confusing since you might mistake a bootboth output file for a bootslickw output file. I get around this by always calling the data files things ending in b if they have both planes involved. Also you can only have a bootslickw or a bootboth running in a directory at one time because both will use Xtemp and Xtemp.slboot. Remember to have the best result at the top of the *.slboot file, just like with bootslickw (see that section for details). The *.slboot files from bootboth look just like the ones from bootslickw.

gridfix

gridfix is a program to a grid search for the best stress tensor with an angular error criterion as described in Michael, 1986. (It uses the 1-cos(beta) error criterion, and an L1 norm). It can be used either with the fault planes picked *a priori*, or with the program picking the preferred fault planes as is done by Gephart and Forsyth (1984), although this is not their error criterion. To run the program you first have to make a modified data file. On each data line you must add a 0,1, or 2 at the end of the line. A 0 will tell the program to pick the preferred plane, a 1 will tell the program to use the given plane, and a 2 will tell the program to use the auxiliary plane as the fault plane. A sample data file is in Appendix I and in the file dixieg. After modifying the data file you then modify the file "controls". This is a include file that tells the program how to search the grid. A sample is in Appendix J.

To compile: % cc -o gridfix -O gridfix.c -lm
To run: % gridfix input_file
Output files: input_file.goput
e.g.: % gridfix dixieg

This will make an output file dixieg.goput, which is shown in Appendix K.

gridstrap

gridstrap is a fast version of gridfix for use in the bootstrap analysis. It uses the same file "controls" to control the grid to be searched, so edit "controls" first.

To compile: % cc -o gridstrap -O gridstrap.c -lm
To run: % gridstrap input_file
Output file: input_file.gboot
e.g.: % gridstrap dixieg

will make a file called dixieg.gboot that is shown in Appendix L.

bootgrid

bootgrid is the analog to bootslickw for the grid search method. It controls the bootstrap resampling process, using gridstrap to analyze the results. To use it first compile gridstrap with the proper grid parameters. Since you will be doing many gridstraps it is best to use a coarse grid limited to the area where the result will be. The more limited and coarser the grid, the faster things will run.

To compile: % make -f makebtgr
To run: % bootgrid input_file n
where n is the number of times to resample the data.
Output files: Xtemp Xtemp.gboot
e.g.: % bootgrid dixieg 2000
will bootstrap resample the dixieg data set 200 times
and run gridstrap on each resampling.
Xtemp.gboot will hold the results and Xtemp the data for the
last resampling.

An example of the output with 10 resamplings is shown in Appendix M. Remember to compute confidence intervals the best result must be at the top of the output file before the resamplings. See bootslickw for details.

plotbootg

plotbootg is the analog of plotboots for the grid search results.

To compile: make -f makeplbtg
To run: plotbootg *.gboot output_file confidence level
Output files: as named in command line.
e.g.: % plotbootg dixieg.gboot dixieg.bplotg 95
will make an output file with the 95% confidence regions for the
dixieg data, based on the resamplings in dixieg.gboot.

dixieg.gboot is an onnet input file, an example is in Appendix N.

onnet

onnet, together with stnet and sttics, comprise a stereonet plotting package. You will have to modify onnet to work with your graphics system. It is a very useful program if you are going to work with stereonets, so it is worth the time. Separate documentation is supplied for onnet, and the code and makefiles are supplied in the directory named ONNET.

Some notes about subroutines

Most of my subroutines are pretty good, the exception may be the eigenvalue-vector routine. I don't use it any more. I wrote it because I wanted to learn how they worked, and it is a bit slow. At the moment I am using the Eispack, but I can't send it around due to license restrictions. I would suggest finding a good eigenvalue routine to use. It will save you time and the answers will be slightly more accurate.

A

first line is a comment

Dixie Valley Slip Data

one datum per line

```
100 45 -112
60 45 -104
60 45 -121
115 47 -108
147 47 -85
165 55 -72
135 50 -95
137 48 -92
140 43 -83
150 52 -86
145 50 -87
170 56 -72
138 50 -85
180 55 -64
135 80 -89
100 48 -102
90 48 -115
85 47 -122
100 42 -109
80 43 -126
105 42 -105
130 65 -81
145 65 -63
115 45 -104
```

dip direction of fault plane (°E of N)

dip of fault plane

rake of fault
as defined by Kanamori
see documentation + Michael (1984)
These are all close to normal faulting which would be 270° or -90°

B

Jul 7 17:10 1986 dixie.oput Page 1

Dixie Valley Slip Data ← Comment line from data file

COORDINATES ARE EAST,NORTH,UP. ← just a reminder

stress tensor is:

0.752	-0.518506	-0.286591
-0.518506	0.360626	-0.017118
-0.286591	-0.017118	-1.11263

best stress in tensor format

eigenvalue	vector: E	N	UP	direction	plunge
1.13354	-0.827207	0.552685	0.101332	123.666170	5.812040
0.0284948	-0.536944	-0.830657	0.147313	32.857183	8.465642
-1.16203	0.16559	0.0674488	0.983885	-112.087886	79.647293

in principal coordinates

variance= 0.0133015 ← $\dfrac{(\text{misfit squared + summed})}{(3 \cdot \text{number of data} - 5)}$

phi value= 0.481381 ← see Michael (1984)

dip direction,	dip,	rake,	fit angle,	mag tau
100.0	45.0	-112.0	3.3	1.07
60.0	45.0	-104.0	3.9	0.69
60.0	45.0	-121.0	13.1	0.69
115.0	47.0	-108.0	6.4	1.13
147.0	47.0	-85.0	1.3	1.05
165.0	55.0	-72.0	0.7	0.89
135.0	50.0	-95.0	5.6	1.11
137.0	48.0	-92.0	3.2	1.10
140.0	43.0	-83.0	5.7	1.06
150.0	52.0	-86.0	6.1	1.03
145.0	50.0	-87.0	3.5	1.06
170.0	56.0	-72.0	3.0	0.84
138.0	50.0	-85.0	2.6	1.10
180.0	55.0	-64.0	5.2	0.73
135.0	80.0	-89.0	14.7	0.57
100.0	48.0	-102.0	7.0	1.10
90.0	48.0	-115.0	2.5	1.04
85.0	47.0	-122.0	8.5	0.99
100.0	42.0	-109.0	0.4	1.03
80.0	43.0	-126.0	12.6	0.89
105.0	42.0	-105.0	1.6	1.06
130.0	65.0	-81.0	8.4	0.98
145.0	65.0	-63.0	13.0	0.92
115.0	45.0	-104.0	2.3	1.12

data listed with angular misfit (β) and magnitude of tangential traction on fault ($|\vec{\tau}|$).

fit angle mean= 5.602842 standard deviation= 4.183697 ($\bar{\beta}$ and standard dev.)

for f=0.8 I= -3.348892 , std. dev.= 1.286381 D norm= 0.317032 ← see Michael (1984) or ignore

avg tau= 0.969106 , std. dev.= 0.157258

← $|\vec{\tau}|$ averaged + std. dev.

C

variance	σ_{11}	σ_{12}	σ_{13}	σ_{22}	σ_{23}	σ_{33}
0.0133015	0.752	-0.518506	-0.286591	0.360626	-0.017118	-1.11263
0.481381	-112.1	79.6	32.9	8.5	123.7	5.8
ϕ	trend	plunge				
	compressional axis		intermediate axis		extensional axis	

D

result for true data

```
0.0133015 0.752 -0.518506 -0.286591 0.360626 -0.017118 -1.11263
0.481381 -112.1   79.6   32.9    8.5  123.7    5.8
0.0121314 0.718949 -0.641149 -0.438707 0.315214 0.0871255 -1.03416
0.575312 -101.5   75.4   33.0   10.3  124.8   10.1
0.0101992 0.773285 -0.468325 -0.163086 0.376979 -0.0682299 -1.15026
0.444264 -126.2   82.7   32.8    6.7  123.1    2.5
0.0163734 0.722273 -0.539473 -0.0898624 0.433902 -0.0874481 -1.15617
0.478776 -138.1   84.0   37.4    6.0  127.4    0.5
0.00823731 0.643188 -0.451217 0.012885 0.541994 -0.134013 -1.18518
0.403354 -170.9   85.3   42.2    3.9  -47.9    2.5
0.0054125 0.649311 -0.405071 -0.0368423 0.517306 -0.127136 -1.16662
0.373271 -155.2   84.6   40.7    5.1  -49.4    1.4
0.0151809 0.7878 -0.551925 -0.259244 0.338372 -0.0159114 -1.12617
0.507155 -113.5   80.6   32.5    7.8  123.2    5.2
0.0142472 0.744277 -0.459018 -0.157031 0.431576 0.00953405 -1.17585
0.428377 -102.0   84.9   35.1    3.7  125.2    3.4
0.0184757 0.776261 -0.507745 -0.376124 0.32838 -0.0238117 -1.10464
0.47514 -112.1   76.9   30.1   10.4  121.5    7.8
0.00973172 0.750847 -0.52625 -0.351678 0.349707 0.0827431 -1.10055
0.49907 -93.0   79.4   32.7    6.2  123.5    8.5
0.0126168 0.830357 -0.505082 -0.400322 0.228312 -0.031177 -1.05867
0.509288 -114.8   75.8   26.3   11.1  117.9    8.6
```

10 results for bootstrap resampled data.

each 2 lines is in the same format as Appendix C

E

plotbootso input file

confidence level

onnet file

```
title dixie.slboot 80 %
r 8.
size line 6
symbol line 3 3
l 123.7   5.8
symbol line 2 2
l  32.9   8.5
symbol line 1 1
l -112.1  79.6
size line 3
# phirange = 0.428377 0.575312
symbol line 1 1
l -101.5  75.4
l -126.2  82.7
l -138.1  84.0
l -113.5  80.6
l -102.0  84.9
l -112.1  76.9
l -93.0  79.4
l -114.8  75.8
symbol line 2 2
l  33.0  10.3
l  32.8   6.7
l  37.4   6.0
l  32.5   7.8
l  35.1   3.7
l  30.1  10.4
l  32.7   6.2
l  26.3  11.1
symbol line 3 3
l 124.8  10.1
l 123.1   2.5
l 127.4   0.5
l 123.2   5.2
l 125.2   3.4
l 121.5   7.8
l 123.5   8.5
l 117.9   8.6
```

best axes

comment line, not plotted but shows range of ϕ with 80% confidence

80% confidence ~~with~~ for S_1 axis

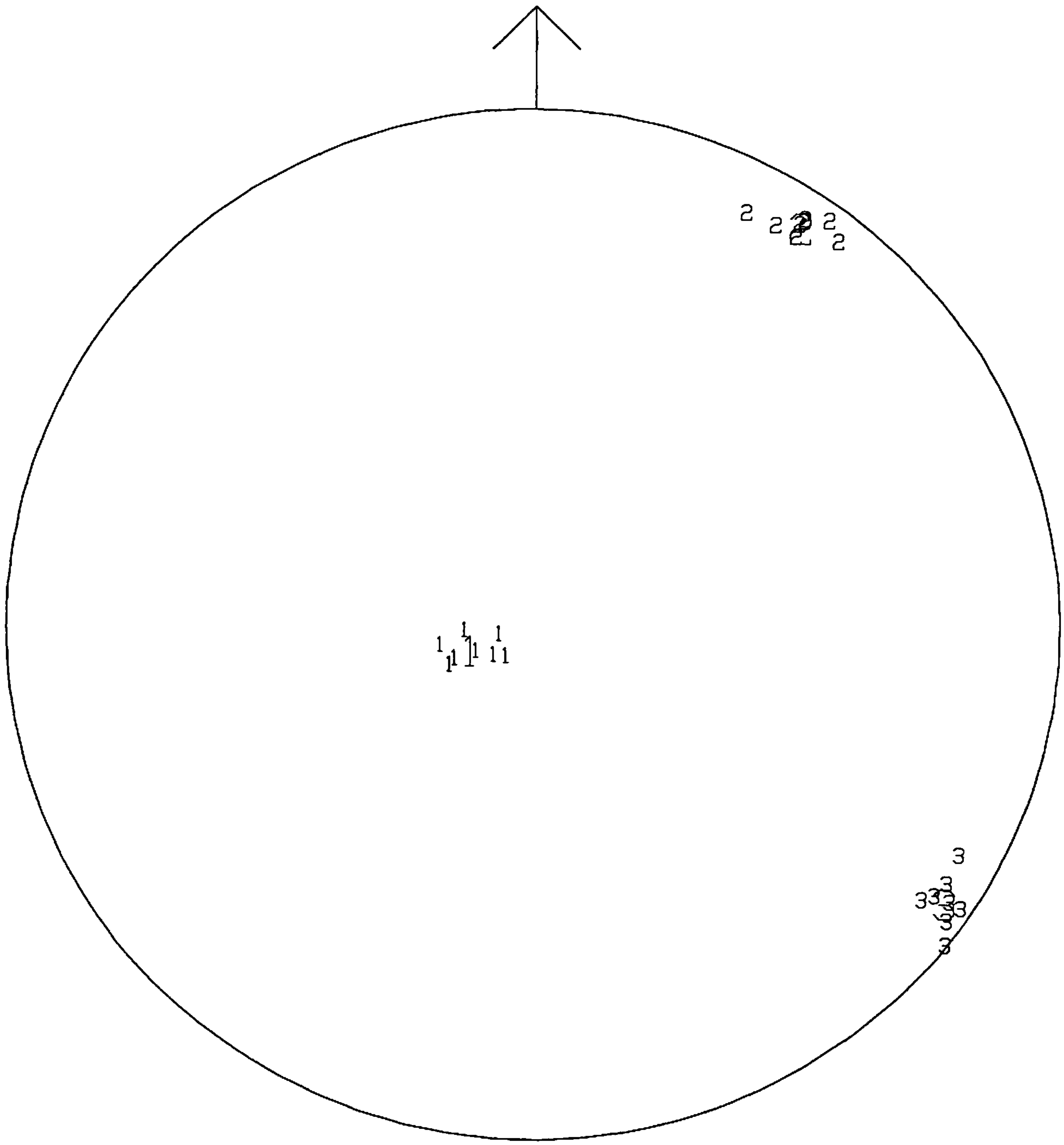
80% conf. for S_2

80% conf. for S_3

E

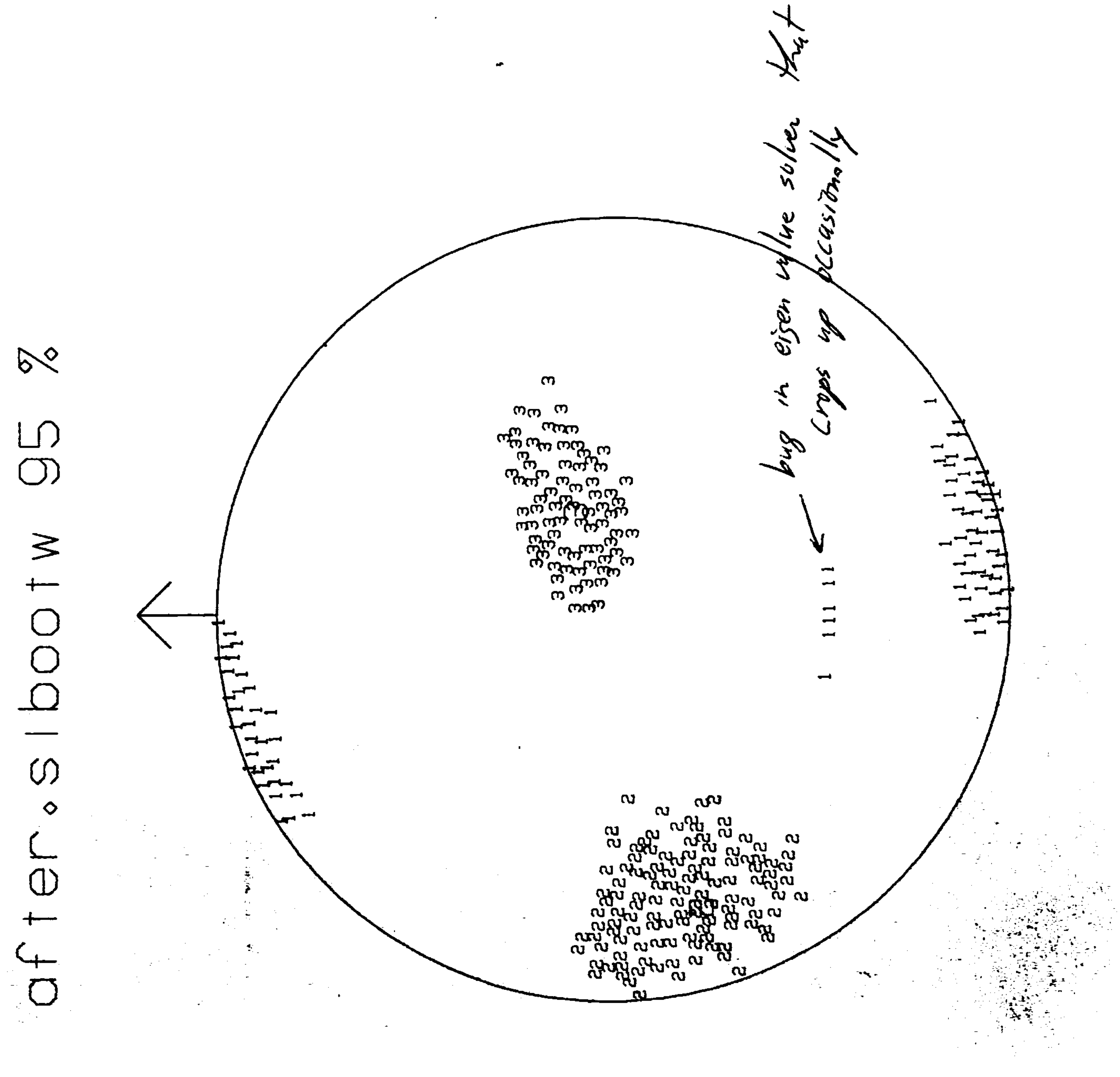
onnet plot of previous page

dixie.slboot 80 %



andy, Mon Jul 7 18:30

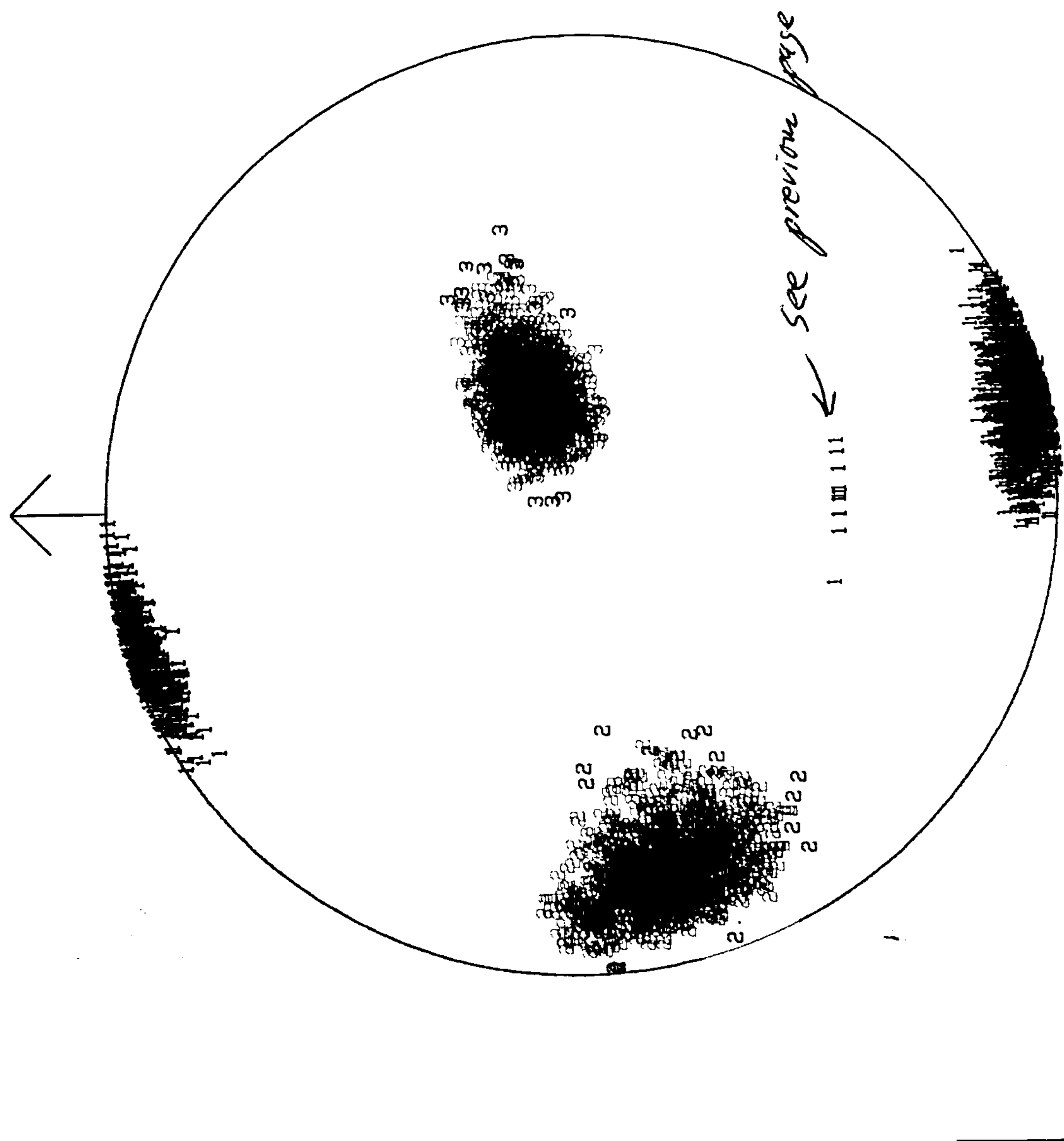
plot boots output E



E

plotboot30 output

after.slbootw 95 %



F

title of plot

onset viewing parameter

plotting commands

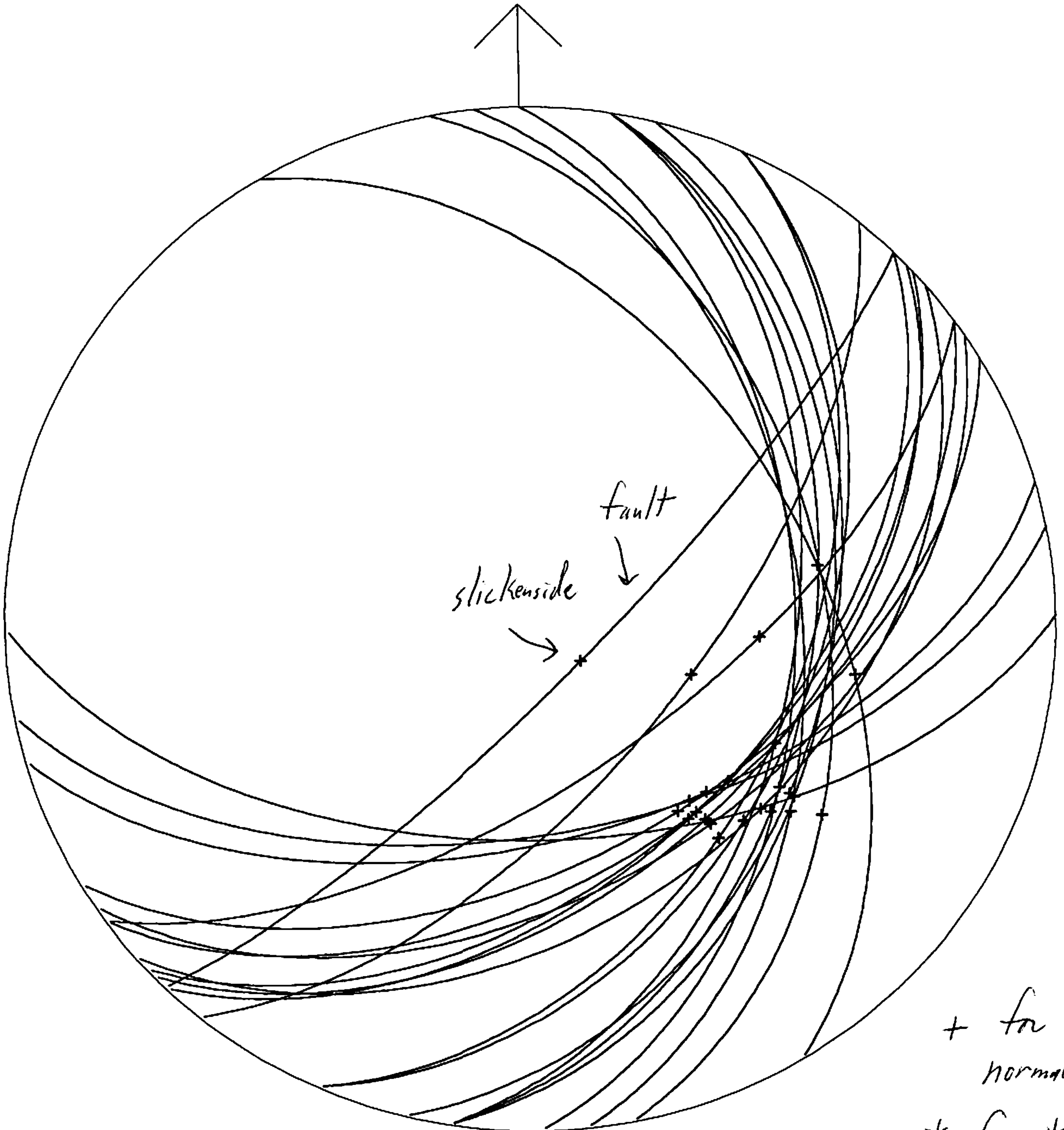
```
title dixie data
fatness plane 1
fatness rake 2
r 8.
plane     100.00     45.00
rake     -112.00
plane      60.00     45.00
rake     -104.00
plane      60.00     45.00
rake     -121.00
plane     115.00     47.00
rake     -108.00
plane     147.00     47.00
rake      -85.00
plane     165.00     55.00
rake      -72.00
plane     135.00     50.00
rake      -95.00
plane     137.00     48.00
rake      -92.00
plane     140.00     43.00
rake      -83.00
plane     150.00     52.00
rake      -86.00
plane     145.00     50.00
rake      -87.00
plane     170.00     56.00
rake      -72.00
plane     138.00     50.00
rake      -85.00
plane     180.00     55.00
rake      -64.00
plane     135.00     80.00
rake      -89.00
plane     100.00     48.00
rake     -102.00
plane      90.00     48.00
rake     -115.00
plane      85.00     47.00
rake     -122.00
plane     100.00     42.00
rake     -109.00
plane      80.00     43.00
rake     -126.00
plane     105.00     42.00
rake     -105.00
plane     130.00     65.00
rake      -81.00
plane     145.00     65.00
rake      -63.00
plane     115.00     45.00
rake     -104.00
```

F

andy, Mon Jul 7 17:21

dixie data

data plot



\+ for normal fault

* for thrust fault

(see figure captions in Michael (1984)

G

Dixie Valley Slip Data FLIPPED denotes data run through switcher

auxiliary planes of dixie data

309.74	49.03	-69.46
259.42	46.68	-76.40
280.36	52.69	-62.75
320.47	45.93	-71.67
319.69	43.23	-95.34
315.47	38.83	-113.81
322.75	40.26	-84.07
319.99	42.04	-87.78
310.47	47.40	-96.48
323.52	38.18	-95.10
320.34	40.09	-93.57
319.84	37.96	-114.61
310.25	40.26	-95.93
319.62	42.59	-122.05
309.26	10.05	-95.65
297.62	43.37	-77.00
304.87	47.66	-64.86
307.50	51.67	-60.39
304.86	50.75	-73.66
304.81	56.51	-61.27
304.83	49.73	-76.88
289.46	26.47	-108.55
274.67	36.15	-134.23
314.42	46.68	-76.40

H

— denotes data has gone through both planes

Dixie Valley Slip Data BOTHWAYS

100.00	45.00	-112.00	original data
309.74	49.03	-69.46	auxiliary plane
60.00	45.00	-104.00	2nd FPS
259.42	46.68	-76.40	
60.00	45.00	-121.00	
280.36	52.69	-62.75	
115.00	47.00	-108.00	
320.47	45.93	-71.67	
147.00	47.00	-85.00	
319.69	43.23	-95.34	
165.00	55.00	-72.00	
315.47	38.83	-113.81	
135.00	50.00	-95.00	
322.75	40.26	-84.07	
137.00	48.00	-92.00	
319.99	42.04	-87.78	
140.00	43.00	-83.00	
310.47	47.40	-96.48	
150.00	52.00	-86.00	
323.52	38.18	-95.10	
145.00	50.00	-87.00	
320.34	40.09	-93.57	
170.00	56.00	-72.00	
319.84	37.96	-114.61	
138.00	50.00	-85.00	
310.25	40.26	-95.93	
180.00	55.00	-64.00	
319.62	42.59	-122.05	
135.00	80.00	-89.00	
309.26	10.05	-95.65	
100.00	48.00	-102.00	
297.62	43.37	-77.00	
90.00	48.00	-115.00	
304.87	47.66	-64.86	
85.00	47.00	-122.00	
307.50	51.67	-60.39	
100.00	42.00	-109.00	
304.86	50.75	-73.66	
80.00	43.00	-126.00	
304.81	56.51	-61.27	
105.00	42.00	-105.00	
304.83	49.73	-76.88	
130.00	65.00	-81.00	
289.46	26.47	-108.55	
145.00	65.00	-63.00	
274.67	36.15	-134.23	
115.00	45.00	-104.00	
314.42	46.68	-76.40	

first focal mechanism

I

Dixie Valley Slip Data

dixie data for grid search

100 45 -112 0

60 45 -104 0

denotes program chooses which plane

60 45 -121 0
115 47 -108 0
147 47 -85 0
165 55 -72 0

1 would be use this plane

135 50 -95 0
137 48 -92 0

2 " " use auxiliary plane

140 43 -83 0
150 52 -86 0
145 50 -87 0
170 56 -72 0
138 50 -85 0
180 55 -64 0
135 80 -89 0
100 48 -102 0
90 48 -115 0
85 47 -122 0
100 42 -109 0
80 43 -126 0
105 42 -105 0
130 65 -81 0
145 65 -63 0
115 45 -104 0

J

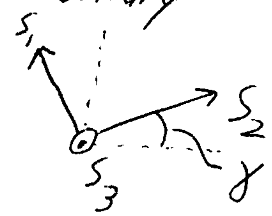
Jul 7 16:08 1986 controls Page 1

```
/* this file contains the control defs for grid.c */
#define THETASTART 0   /*THETA is the trend of S1, range is 0 to 360 */
#define THETASTOP 360
#define THETASTEP 10
#define BETASTART 0 /* BETA is the plunge of S1, range is 0 to 90 */
#define BETASTOP 60
#define BETASTEP 10
#define GAMMASTART -40   /* GAMMMA is the rotation angle, range is 0 to 180 */
#define GAMMASTOP 40
#define GAMMASTEP 10
#define PHISTART 0  /*PHI is phi, range is 0 to 1 */
#define PHISTOP 1
#define PHISTEP .2
```

Theta – trend of extensional axis (S_3) °E of N

Beta – plunge " " "

Gamma – rotation of S_2 (intermediate axis w.r.t. an arbitrary frame about S_3)



Phi – see Michael (1984)

- - - - = arbitrary coordinates picked by program

Start – starting value for grid

Stop – stopping " " "

Step – step size

In this case we search for $0^\circ \le \theta \le 360^\circ$ in 10° steps.

K

```
/* this file contains the control defs for grid.c */
#define THETASTART 0   /*THETA is the trend of S1, range is 0 to 360 */
#define THETASTOP 360
#define THETASTEP 10
#define BETASTART 0 /* BETA is the plunge of S1, range is 0 to 90 */
#define BETASTOP 60
#define BETASTEP 10
#define GAMMASTART -40   /* GAMMMA is the rotation angle, range is 0 to 180 */
#define GAMMASTOP 40
#define GAMMASTEP 10
#define PHISTART 0  /*PHI is phi, range is 0 to 1 */
#define PHISTOP 1
#define PHISTEP .2
Dixie Valley Slip Data
 100.0   45.0 -112.0    0
  60.0   45.0 -104.0    0
  60.0   45.0 -121.0    0
 115.0   47.0 -108.0    0
 147.0   47.0  -85.0    0
 165.0   55.0  -72.0    0
 135.0   50.0  -95.0    0
 137.0   48.0  -92.0    0
 140.0   43.0  -83.0    0
 150.0   52.0  -86.0    0
 145.0   50.0  -87.0    0
 170.0   56.0  -72.0    0
 138.0   50.0  -85.0    0
 180.0   55.0  -64.0    0
 135.0   80.0  -89.0    0
 100.0   48.0 -102.0    0
  90.0   48.0 -115.0    0
  85.0   47.0 -122.0    0
 100.0   42.0 -109.0    0
  80.0   43.0 -126.0    0
 105.0   42.0 -105.0    0
 130.0   65.0  -81.0    0
 145.0   65.0  -63.0    0
 115.0   45.0 -104.0    0
6.245062    0     0  -40 0.00
NNNNNNNNNNNNNRNNNNNNNRRN
5.010866    0     0  -40 0.20
NNNNNNNNNNNNNRNNNNNNNRNN
3.093110    0     0  -30 0.00
NNNNNNNNNNNNNRNNNNNNNNNN
3.020180    0     0  -30 0.20
NRNNNNNNNNNNNRNNNNNNNNRN
2.510632    0     0  -20 0.00
NNNNNNNNNNNNNRNNNNNNNNNN
2.377645    0     0  -20 0.20
NRNNNNNNNNNNNRRNNNNNNRRN
1.043870    0     0  -10 0.00
NNNNNNNNNNNNNNNNNNNNNNNN
0.734450    0    10    0 0.00
NRRNNNNNNNNNNNNNNNRRNNNN
0.673273   30     0  -10 0.00
NNNNNNNNNNNNNNNNNNNNNNNN
```

repeat controls

data

lowest result so far

new best result

$\Sigma 1-\cos(\beta)$ θ β γ ϕ

N means given plane contributes to sum

R " auxiliary " " " "

k

```
0.610735   40     0   -10 0.00
NNNNNNNNNNNNNNNNNNNNNNNN
0.581994   50     0   -10 0.00
NNNNNNNNNNNNNNNNNNNNNNNN
0.563105   80     0     0 0.20
RNRRNNRRNNNNNNRRRRRRRNNR
0.521476   90     0     0 0.20
RNNRNNRRNNNNNNRRRRRRRNNR
0.504014  100     0     0 0.20
RNNRNNRRNNNNNNRRRRRRRNNR
0.466772  100     0     0 0.40
RNNRNNRRNNNNNNRRRRRRRNNR
0.415258  110     0   -10 0.20
RNNRNNNNNNNNNNRNRRRNNNNR
0.409022  110     0     0 0.40
RNNRNNRRNNNNNNRNNNRRRNNR
0.357269  110    10   -20 0.40
NNNNNNNNNNNNNNNRNRNRNNNN
0.343252  110    10   -10 0.40
RNNNNNNNNNNNNNNRRRRRNNNN
0.248419  120     0   -10 0.40
NNNRNNNNNNNNNNNNRNNNNNNN
0.193974  120    10   -20 0.60
RNNNNNNNNNNNNNNRNNRNRNNN
0.189476  130     0   -10 0.40
NNNNNNNNNNNNNNNRNNNNNNNR
0.189476  310     0    10 0.40  final best
NNNNNNNNNNNNNNNRNNNNNNNR
                                  θ        β         γ         φ
best,theta,gamma,beta,phi=  0.189476 310.000000 0.000000 10.000000 0.400000
s3= -0.766044 0.642788 0.000000  extensional axis
s2= -0.633022 -0.754407 0.173648  intermediate axis
s1= 0.111619 0.133022 0.984808  compressional axis
N  first fault given plane picked
N  2nd fault " " "
N
N
N
N
N
N
N
N
N
N
N
N
N
R
N
N
N
N
N
N
N
R  24th fault auxiliary plane picked
```

L


```
                                  Σ1-cos β      θ          β          γ          φ
best,theta,gamma,beta,phi=  0.189476 310.000000 0.000000 10.000000 0.400000
s3= -0.766044 0.642788 0.000000
s2= -0.633022 -0.754407 0.173648   best axes
s1= 0.111619 0.133022 0.984808
0.587842 -0.41174 -0.131908
-0.41174 0.44264 -0.157202         best tensor
-0.131908 -0.157202 -1.03048
```

M


```
best,theta,gamma,beta,phi=  0.189476 310.000000 0.000000 10.000000 0.400000
s3= -0.766044 0.642788 0.000000
s2= -0.633022 -0.754407 0.173648
s1= 0.111619 0.133022 0.984808
0.587842 -0.41174 -0.131908
-0.41174 0.44264 -0.157202
-0.131908 -0.157202 -1.03048
```

best result for true data

```
best,theta,gamma,beta,phi=  0.085234 300.000000 10.000000 10.000000 0.200000
s3= -0.852869 0.492404 -0.173648
s2= -0.518518 -0.837792 0.171010
s1= -0.061275 0.235889 0.969846
0.684946 -0.144856 0.154323
-0.144856 0.407955 -0.400243
0.154323 -0.400243 -1.0929
best,theta,gamma,beta,phi=  0.056462 140.000000 10.000000 0.000000 0.600000
s3= 0.633022 -0.754407 -0.173648
s2= 0.766044 0.642788 0.000000
s1= 0.111619 -0.133022 0.984808
0.33756 -0.561189 -0.219846
-0.561189 0.535466 0.262003
-0.219846 0.262003 -0.873026
```

best results for resample data

```
best,theta,gamma,beta,phi=  0.104157 310.000000 0.000000 10.000000 0.400000
s3= -0.766044 0.642788 0.000000
s2= -0.633022 -0.754407 0.173648
s1= 0.111619 0.133022 0.984808
0.587842 -0.41174 -0.131908
-0.41174 0.44264 -0.157202
-0.131908 -0.157202 -1.03048
best,theta,gamma,beta,phi=  0.058090 270.000000 0.000000 20.000000 0.200000
s3= -1.000000 0.000000 0.000000
s2= -0.000000 -0.939693 0.342020
s1= 0.000000 0.342020 0.939693
0.8 -7.00187e-09 -6.13213e-09
-7.00187e-09 0.212836 -0.51423
-6.13213e-09 -0.51423 -1.01284
best,theta,gamma,beta,phi=  0.055306 310.000000 10.000000 10.000000 0.200000
s3= -0.754407 0.633022 -0.173648
s2= -0.656121 -0.735024 0.171010
s1= -0.019382 0.242945 0.969846
0.627051 -0.183488 0.0824773
-0.183488 0.465851 -0.420961
0.0824773 -0.420961 -1.0929
best,theta,gamma,beta,phi=  0.083994 300.000000 10.000000 10.000000 0.200000
s3= -0.852869 0.492404 -0.173648
s2= -0.518518 -0.837792 0.171010
s1= -0.061275 0.235889 0.969846
0.684946 -0.144856 0.154323
-0.144856 0.407955 -0.400243
0.154323 -0.400243 -1.0929
best,theta,gamma,beta,phi=  0.103263 300.000000 10.000000 10.000000 0.200000
s3= -0.852869 0.492404 -0.173648
s2= -0.518518 -0.837792 0.171010
s1= -0.061275 0.235889 0.969846
0.684946 -0.144856 0.154323
-0.144856 0.407955 -0.400243
0.154323 -0.400243 -1.0929
```

M

```
best,theta,gamma,beta,phi=  0.110173 130.000000 20.000000 -10.000000 0.800000
s3= 0.719846 -0.604023 -0.342020
s2= 0.587526 0.792582 -0.163176
s1= 0.369641 -0.083484 0.925417
0.374432 -0.683342 -0.530752
-0.683342 0.180962 0.361444
-0.530752 0.361444 -0.555394
best,theta,gamma,beta,phi=  0.143403 310.000000 0.000000 0.000000 0.400000
s3= -0.766044 0.642788 0.000000
s2= -0.642788 -0.766044 0.000000
s1= 0.000000 0.000000 1.000000
0.602793 -0.393923 0
-0.393923 0.463874 0
0 0 -1.06667
```

N

```
title dixieg.gboot 80 %
r 8.
size line 6
symbol line 3 3
l -50.0   0.0
symbol line 2 2
l  40.0  10.0
symbol line 1 1
l -139.9  79.9
size line 3
# phirange = 0.2 0.6
symbol line 1 1
l 165.3  75.8
l -40.0  79.9
l -139.9  79.9
l 179.9  70.0
l 175.3  75.8
l 165.3  75.8
l 165.3  75.8
l  89.9  89.9
symbol line 2 2
l  31.7   9.8
l  50.0   0.0
l  40.0  10.0
l   0.0  20.0
l  41.7   9.8
l  31.7   9.8
l  31.7   9.8
l -139.9   0.0
symbol line 3 3
l -60.0  10.0
l 139.9  10.0
l -50.0   0.0
l -89.9   0.0
l -50.0  10.0
l -60.0  10.0
l -60.0  10.0
l -50.0   0.0
```

Just like Appendix E
but for grid search

andy, Mon Jul 7 19:33

dixieg.gboot 80 %

Just like E
but for
grid search