

DEFNODE User's Manual

Version 2007.10.25

Author: [Rob McCaffrey](#)

Email: mccafr@rpi.edu or R.McCaffrey@gns.cri.nz

Last webpage update: Oct. 25, 2007

CONTENTS

1. [BACKGROUND](#)
2. [CONTROL FILE](#)
3. [OUTPUT FILES](#)
4. [PLOTING WITH GMT](#)
5. [SAMPLE INPUT](#)
6. [CITATIONS](#)

BACKGROUND

Version dates: 08.12.1995, 03.22.1996, 10.26.1996, 09.28.1999, 12.09.1999, 01.13.2000, 12.27.2000, 10.19.2001, 06.27.2002, 07.16.2002, 08.16.2002, 11.01.2002, 05.01.2003, 06.05.2003, 10.28.2003, 12.31.2003, 02.17.2004, 06.22.2004, 01.07.2005, 08.05.2005, 10.13.2005 01.12.2006, 08.28.2006, 10.25.2007

Link to [Papers using DEFNODE](#) (listed after citations).

DEFNODE is a Fortran program to model elastic lithospheric block rotations and strains, and locking or coseismic slip on block-bounding faults. Block motions are specified by spherical Earth angular velocities (Euler rotation poles) and interseismic backslip is applied along faults that separate blocks, based on the routines of Okada (1985; 1992). The faults are specified by lon-lat-depth coordinates of nodes (forming an irregular grid of points) along the fault planes. The parameters are estimated by simulated annealing or grid search.

DISCLAIMER I make no guarantees whatsoever that this program will do what you want it to do or what you think it is doing. It has more than 20,000 lines of code and I guarantee some bugs are there. I have not tested every option thoroughly and have not documented every option. However, I use it extensively for my own research as is. I am happy to hear about tests you perform and will try to fix any bugs you discover.

REQUEST Please do not make changes to the code and re-distribute it. I am happy to help with any improvements or changes.

The program can solve for

- interseismic plate locking or coseismic slip distribution on faults,
- block (plate) angular velocities,
- uniform strain rates within blocks, and
- rotation of GPS velocity solutions relative to reference frame.

Data to constrain the models include

- GPS vectors,
- surface uplifts,
- earthquake slip vectors,
- spreading rates,
- rotation rates,
- fault slip rates,
- transform azimuths,
- surface strain rates, and
- surface tilt rates.

Output files comprise text files suitable for plotting with [GMT](#) (Wessel and Smith, 1991).

Prior users - changes:

The program is largely backwards compatible, all earlier commands should work except as noted here.

- Version 2006.08.28
 - GP: line requires a pole index number specifying which pole to use in rotating the velocity field.
 - PG: line needs a 'c' (ie, PGc:) if specifying a Cartesian angular velocity.
 - ST: line - strain rates are now in nanostrain/year instead of microstrain/yr.
 - The .pol output file is now called .poles and does not have strain tensors in it, they are in the .strain file.
 - NF: is officially now NN:, but NF: still works.
 - NO: is officially now NV:, but NO: still works.
 - No more WARNING file - all warnings are output to screen.
 - GF: options now incorporated into GD: but GF: still works
- Version 2007.10.25
 - Included files now have .h extension.
 - New options: EI, ES, RC, RF, SN
 - Changes made to options: FA, GD, GR, GS, NI, PO, RM
 - For FA:, see *Changing fault dip azimuth*, the format has changed.
 - For RM: add a 'b' (i.e. RMb:) if you are removing sites from an entire block.

COMPILATION: The program can be compiled with various fortran compilers. Lately I have been using gfortran in Fedora Linux to compile. Other compilers may be fussy and give errors or warnings. Let me know and I can try to fix them.

Files needed:

```
-- source code
defnode.f

-- include files
defconst.h defcomm1.h defcomm2.h deffiles.h [defedc.h defedp.h]

-- Smithsonian volcanoes file (optional, see code for format)
votw.gmt

-- Engdahl et al. earthquakes file (optional, see code for format)
ehb.gmt
```

Download [defnode zip file](#).

Includes an example (data and control file from Costa Rica).

Before compiling, do the following:

1. Set array dimensions within 'defcomm1.h'. The dimensions are set in a PARAMETER statement and all start with MAX_. Be sure to keep the structure of the PARAMETER statement intact. Exceeding array dimensions is not always checked explicitly and can cause strange behavior.

2. [Optional] Set paths to the volcanoes (votw.gmt; needed if +vtw flag set) and earthquakes (needed if +eqk flag set) files in the file 'deffiles.h' (see more instructions in file). For example:

```
** path of volcanoes file
   volcfile = '/data/votw.gmt'

** path of earthquake file
   num_quakefiles = 1
   quakefile(1) = '/data/ehb.gmt'
```

RUNNING: If you type

```
% defnode
the program will ask for the control file name and the model name. Or type the control file name as a command line argument:
```

```
% defnode control_file_name
```

Or also type model name as second command line argument:

```
% defnode control_file_name model_name
```

Runtime messages are all output to the screen. Many files are generated as discussed later.

NOTES:

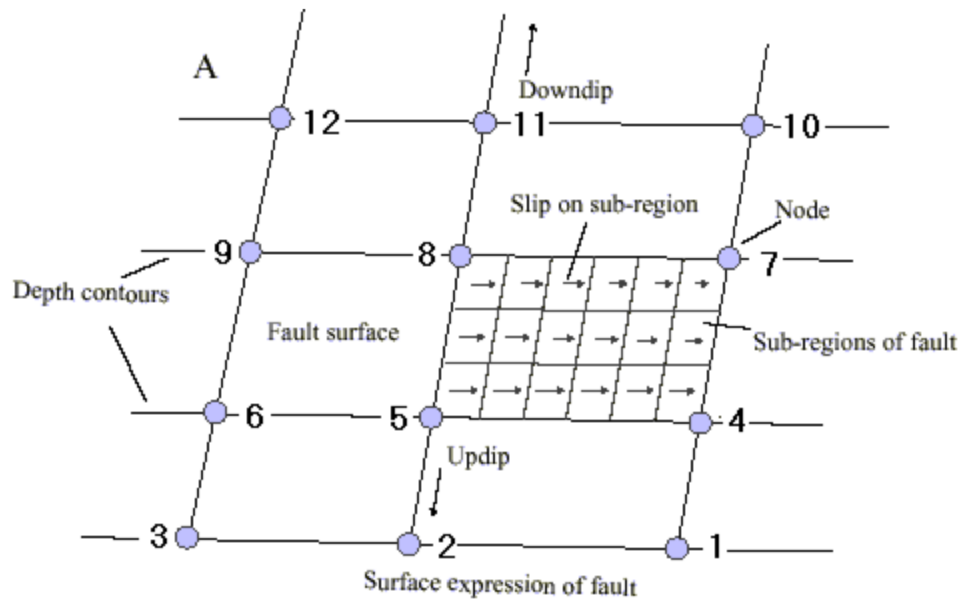
Directories: All output will be put into a directory specified by the MO: (model) command. The program also produces a directory called 'gfs' (or a user-assigned directory) to store the Green's function files.

Poles (angular velocities) and blocks: You can specify many poles and many blocks (dimensioned with MAX_poles, MAX_blocks). There is NOT a one-one correspondence between poles and blocks. More than one block can be assigned the same pole (ie, the blocks rotate together) but each block can be assigned only one pole. Poles can be specified as (lat,lon,omega) or by their Cartesian components (W_x , W_y , W_z).

Strain rates and blocks: The strain rate tensors (SRT) for the blocks are input in a similar way as the rotation poles. Each SRT is assigned an index (integer) and blocks are assigned a SRT index. As with poles, more than one block can be assigned to a single SRT. Velocities are estimated from the SRT using the block's centroid as origin (default) or a user-assigned origin; if multiple blocks use the same SRT assign an origin for this SRT (see ST: option)

Faults and blocks: Faults along which backslip is applied are specified and must coincide point-for-point at the surface with block boundary polygons. However, not all sections of block boundaries have to be specified as a fault. If the boundary is not specified as a fault it is treated as free-slipping and will not produce any elastic strain (ie, there will be a step in velocity across the boundary). By specifying no faults, you can solve for the block rotations alone.

Fault nodes: Fault surfaces are specified in 3 dimensions by nodes which are given by their longitude and latitude (in degrees) and depth (in km, positive down). Nodes are placed along depth contours of the faults and each depth contour has the same number of nodes. Nodes are numbered in order first along strike, then down dip. The figure below shows the numbering system for the nodes. Strike is the direction faced if the fault dips off to your right. Faults cannot be exactly vertical (90° dip) as the hangingwall and footwall blocks must be defined. The fault geometry at depth can be built either by specifying all the node coordinates individually or by using the DD: and ZD: options.



Indexing of nodes on the fault surface.

The coupling fractions (ratio of locked to total slip, called ϕ) or slip amplitude (for coseismic applications) are either specified or estimated at the nodes. The 'slip deficit rate vector' is ϕ is multiplied by the slip vector V at the node, where V is estimated from the angular velocities. The *slip rate deficit* gives rise to the elastic deformation around the fault. For coseismic, ϕ is the fault slip amplitude and the unit vector V gives the slip direction.

The elastic deformation is calculated by integrating over small patches (quadrilaterals) in the regions between the nodes. The Okada method is used to calculate surface velocities while applying backslip at a rate of $-V\phi$ (or $V\phi$ for coseismic) on each of these little patches. Because the Okada formulas used are for rectangular patches, the sizes of the interpolated patches should be kept small (less than a few kilometers). As the patches get smaller their deviations from rectangles matters less (the point source approximation).

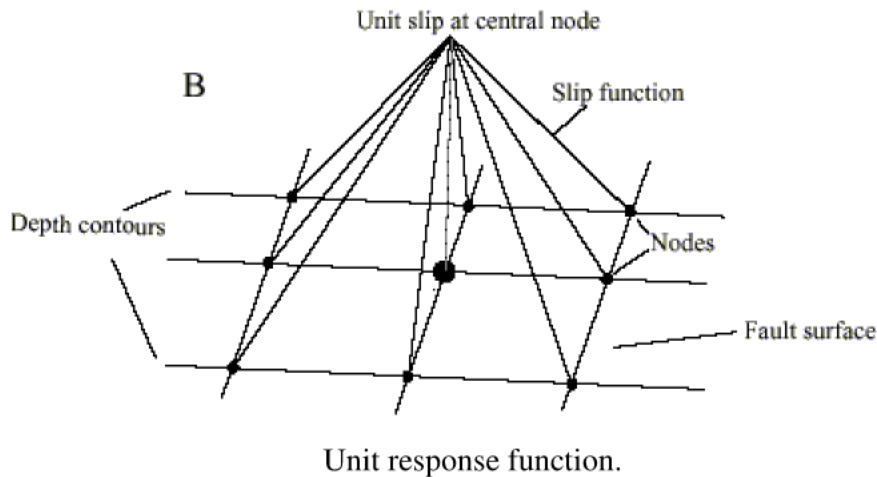
The distribution of slip on the fault can be parameterized in several ways (see FT: option for details). The nodes can be treated as independent parameters or can be grouped such that multiple nodes have the same ϕ value. The distribution of slip can also be set to one of a few specified functions of depth (exponential, boxcar, or Gaussian) along depth profiles, called z-profiles. In this case, the parameters for the function can be varied along strike on the z-profiles. This version allows 2D Gaussian distributions of slip on the fault surface that may be suitable for earthquakes or slow slip events.

Green's functions: If you are performing an inversion, the program uses unit response (Green's) functions (GFs) for the elastic deformation part of the problem since the inversion method (downhill simplex) has to calculate numerous forward models. The GFs are put in a directory called 'gfs' (or a user-specified directory using GD: option) and the files are named with the form gf001001001g, gf001002001g, etc. First 3 digits are fault number, next 3 are the along strike (X) node index, the next 3 are the downdip (Z) node index, the final letter is the data type; g - GPS, u - Uplift, t - Tilts, s - strain rates. Once you have calculated GFs for a particular set of faults you

can use these in inversions without recalculating them (see option GD:). The GFs are based on the node geometry, GPS data, uplift data, strain tensor data, and tilt rate data so if you change the node positions or ADD data, you need to re-calculate GFs. If you REMOVE data, you do not need to recalculate GFs. You can add or subtract slip azimuths, slip rate, and rotation rate data without re-calculating GFs since those data are calculated from the rotation poles only. If you change a node position the program will detect the change it and re-calculate only the necessary GFs.

If you add GPS vectors, the program will not detect that and GFs may not be calculated. In this case, re-calculate all the GFs.

The GFs are the responses at the surface observation points to a unit velocity (or displacement) in the North and East directions at the central node. The slip velocity is tapered to zero at all adjacent nodes.



Data files and weighting: Data files are generally in free-format but the information must be in the correct order as outlined below. Multiple data files can be specified and they are all read in and used. An uncertainty scaling factor F can be applied to each data file; this number is multiplied by the data standard errors given in the file. Since the weight applied is the inverse of the datum variance, the weight of the datum will be multiplied by F^{-2} . If s is the standard deviation of the datum given in the file, the new standard deviation $s' = sF$ and the weight $=s'^{-2} = (sF)^{-2}$. Data covariance is used when the correlation coefficient is given for GPS vectors.

Some data types have the option of being entered within the control file itself.

Inversion: The inversion estimates the set of parameters that minimizes the reduced chi-squared statistic:

$$X_n^2 = [\text{SUM } r^2 / (sF)^2] / dof$$

where r is the residual, s is the standard deviation, and dof is the degrees of freedom. For angular data, the r/s is determined using the equation of DeMets et al. (1990).

Minimization is performed by the simulated annealing technique (see Press et al. 1989). You supply the number of iterations and the initial temperature in the `sa:` command. If `Temp = 0` the downhill simplex method is used.

There is also the option of performing grid searches for the minimum X_n^2 .

Parameter constraints are applied by using penalties for parameters that stray outside the specified bounds.

Units:

- Slip rates, GPS velocities -- millimeters per year (millimeters for coseismic)
- Depths -- kilometers
- Latitude, longitude, azimuths -- degrees
- Strain rates -- nanostrain/year (10^{-9} per year)
- Rotation rates -- degrees per million years

Coordinates:

- Most coordinates are entered in geographic lon, lat order unless noted otherwise (Lat,lon input is allowed in the `BL:` and `FA:` controls.)
- Poles are lat, lon, omega following common use - Cartesian components can also be used.
- North latitude is positive, South is negative. Longitude can be entered in either 0 to 360, or -180 to +180 degrees - the program converts all longitudes to 0 to 360 unless a flag is set to use -180 to +180.

Miscellaneous notes:

- Each fault must have a unique footwall block. It can have a changing hangingwall block (along strike) which the program determines (the hanging wall block you specify does not have to be correct, but the footwall block must be correct and unique for this fault).
- Fault dip cannot be 90° or greater. (Hangingwall and footwall must be unambiguous.)
- Either a profile, grid, or input data are required to tell the program where to calculate surface velocities.
- Exceeding array dimensions is not always checked explicitly and can cause strange behavior.
- It is sometimes advisable to put character strings in quotes (filenames, for example) if the program has trouble reading the file.
- The program tries to catch simple mistakes and produces warnings, output to the screen.
- To stop iterating, create a file called 'stop', 'MODLstop' or 'stopMODL' in the working directory (MODL is the model name given by the `MO:` option).

CONTROL FILE

The program reads the model and all controls from an ASCII file. Some data can be in the control file. The control file format is described here.

Lines in the control file comprise a keyword section and a data section. The keyword section starts with a 2-character keyword (in the first 2 columns) and ends with a colon (:). Normally only the first 2 characters of the keyword are used so in general any characters between the 3rd character and the : are ignored. (Sometimes the third character specifies a format as outlined below.)

THE KEYWORD MUST START IN THE FIRST COLUMN OR THE LINE IS IGNORED.

Case does not matter.

The data section of the input line goes from the colon to the end of the line and its contents depend on the keyword. In a few cases the data section comprises multiple lines (i.e., always BL: and FA:, and sometimes others).

For example, the key characters for a fault are 'FA' and this has two arguments, the fault name and the fault number, so the following lines are correct:

```
fa: JavaTrench 1
fault: JT 1
fault (Java trench): JavaTr 1
FA: JT 1
but
thrust fault: 1
  fault: JT 1
fault 1 : JavaTrench
are not valid.
```

Some notes on input lines:

- Lines without the key characters in the first 2 columns are ignored, unless they are part of a multiline data section.
- Input lines can be commented out by putting any other character in the first column (' or # or a space, for examples).
- Multiline data sections can be skipped by commenting out the keyword line only.
- Contiguous lines of input can be skipped by bracketing them within the SK: (skip) and CO: (continue) options.
- It is advisable and good practice to start comment lines with a space, *, # or some other character outside the range A - Z (the program has many undocumented options and you may trigger one by accident).

Multiple lines can exist in the control file for a particular input; the program uses the last occurrence it finds. For example, if the following pole specification lines are in the control file

pole: 1 -20 40 .2

pole: 1 -19 33 .1

the values (-19, 33, .1) will be assigned to pole 1.

An exception to this rule pertains to data files, that are all used.

The order of the statements in the control file should not matter since the program reads it in 3 times, the first time getting the blocks, then the faults, then the rest. The control file can contain lines after the end of input (EN:) statement and these are ignored.

Summary of key characters:

BL: outline of elastic rotating plate polygon

BP: specify pole and strain tensor indices for a block

CF: connect 2 faults (remove overlap or gap from subsurface intersection of two faults)

CL: clear specified data type

CO: continue reading from input file (used with SK: option)

EI: list which 2D Gaussian sources to adjust

EM: end of model input section

EN: end of input data

EQ: equate two nodes on different faults (set their phi's equal)

ES: 2D Gaussian source (coseismic)

FA: fault geometry input

DD: set depth and dip to nodes (use only within FA: section)

ZD: similar to DD:

FF: fault flags (turn faults on and off)

FL: set flags

FS: calculate and output relative block velocities at specified points

FT: fault parameterization type

FX: specify position of a particular fault node - overrides all other specifications

GD: specify Green's functions directory and other GF parameters

GI: GPS velocity field rotations (relative to reference frame) to be adjusted

GP: GPS input data file

GR: grid of vectors to calculate

GS: parameter grid search controls

HC: hard constraints to apply

IN: interpolation lengths for fault segments between nodes (for final forward run)

MF: merge faults at T-junction

MM: range of seismic moments allowed for a fault

MO: model experiment name, used for output filenames

MV: move surface points

NI: number of iterations

NN: node parameter index numbers (same as old NF:)

NV: node values (same as old NO:)

NX: indices of fixed nodes

PE: scaling factors for penalty functions

PF: parameter and model I/O file

PG: initialize pole of rotation for GPS vector file

PI: block poles to be adjusted

PM: parameter min and max values allowed

PN: node z-profile parameter index numbers

PO: block pole of rotation values
PR: surface profile line
PV: node z-profile parameter values
PX: fix node z-profile parameters
RC: remove sites within a specified circular area (e.g., volcanic region)
RE: reference block for vectors
RF: rotate reference frame for vector output
RM: remove named GPS sites or blocks from data
RO: rotation rates input data file
RS: reference site for GPS vectors
SA: simulated annealing inversion controls
SI: strain rates tensors to be adjusted
SK: skip following lines of input data until a CO: line is encountered
SM: apply along-strike smoothing to fault phi
SN: snap block boundary points together
SR: fault slip rate / spreading rate data file
SS: strain rate tensor data file
ST: initialize strain rate tensor values
SV: slip vector / transform azimuth data file
TI: tilt rate data file
UP: uplift rate data file

Descriptions of Key Characters and input format:

Key characters and formats. Examples are given at bottom. Brackets { } show optional inputs.
MODL = 4-char model name.

BL - block (plate) outline

BL: NAME M N {optional filename}
[multiline data section]

NAME = 4-character name of block

M = Rotation pole index for block (overridden by BP: option)

N = Strain rate tensor index for block (overridden by BP: option)

filename = optional file containing block outline

Multiline data section (alternatively, contents of filename)

First line: Number of corners outlining block, { CentroidX CentroidY }

CentroidX = x coordinate of block centroid (optional)

CentroidY = y coordinate of block centroid (optional)

For each corner, one coordinate pair (lon, lat) in each line

```
b1: NOAM 1 1
4 50 50
-135 55
-130 44
-100 44
```

```
260 55
```

or

```
bl: NOAM 1 1 NOAM.block
```

where NOAM.block is a file containing:

```
4 50 50
-135 55
-130 44
-100 44
260 55
```

- If the block centroid is not given (or 0, 0), it is calculated. These centroids are used when solving for strain rates in block and are output to files (for labels in maps).
- Don't close the block by making first and last points the same - the program does this for you.
- To avoid counting corners of the block, use 9999 for the number of corners, and 9999, 9999 for the last corner. For example:

```
bl: NOAM 1 1
9999 50 50
-135 55
-130 44
-100 44
260 55
9999 9999
```

BP - specify pole/strain tensor for block

```
BP: NAME N M
```

Block NAME uses pole of rotation N and strain rate tensor M. This overrides the assignments given in BL: option. Set to 0 if no pole or strain tensor is used.

```
bp: NoAm 1 0
bp: JdFa 2 0
bp: EOre 3 1
```

CF - connect faults

```
CF: F1 F2
```

Connect 2 faults at their intersection. Where fault #F1 and fault #F2 intersect at the surface, force deeper nodes to also intersect by moving nodes (at same depth) from both faults to the average position of the two nodes. Both faults must have their nodes at the same depths for this to work.

```
cf: 5 7
```

CL - clear data

Remove data/controls read in thus far.

- 'gps' for GPS vectors
- 'svs' for slip vectors
- 'srs' for fault slip rates
- 'sss' for surface strain rates
- 'srs' for fault slip rates
- 'ups' for uplift rates
- 'tlt' for tilt rates
- 'rot' for rotation rates
- 'nrm' turn off all RM: lines
- 'hcs' turn off all hard constraints

cl: gps svs
Up to 8 per line.

CO - continue

CO:
Continue reading data from file (turns off SK: skip mode).

DD - specify dip and depth of fault segment

see FA: option

EI - 2D Gaussian sources to adjust

EI: F N N N
For fault F, adjust the 2D Gaussian slip sources listed by N.
ei: 2 1 2 3

EM - end of model section

EM:
End of model input section (no arguments)

To specify unique parameters for several models in a single input file, make a model input section as follows:

```
mo: mod1  
pi: 1 2 4  
gi: 1  
pf: "mod1/io1.pio" 3
```

```
mo: mod2  
pi: 2 3  
gi: 2  
pf: "mod2/io2.pio" 3
```

```
mo: mod3  
rm: INDO BABI  
pi: 3  
pf: "mod3/io3.pio" 3
```

em:

then when you run defnode you specify the model to use as the second command-line argument, for example:

```
% defnode models.inp mod2
```

The first MO: command marks the beginning of the models input section, and the EM: command marks its end.

If you do not specify a model (in the second argument) defnode will use the last model it finds in the file (the MO: - EM: structure is ignored). In the case above it will run model mod3 but will use some of the specifications of the prior models, such as the GI: line which is not given for mod3.

For a specified model, the program will process all input lines leading up to the first MO: line, all lines within the specified model, and all lines after the EM: line up to the EN: line.

EN - end of data in control file

EN:

End of control file (no arguments), anything in the file after this line is ignored.

EQ - equate two nodes

EQ: F1 X1 Z1 F2 X2 Z2

Make two nodes on different faults have same value of ϕ in the inversion. F, X, and Z are the indices for the nodes.

```
eq: 1 4 2 2 1 2
```

forces the node of the first fault which is fourth along strike and second downdip to have the same ϕ as the second fault's first node along strike and second downdip.

ES - 2D Gaussian slip source

ES: Flt# Srce# Long Lat X-dim W-dim Amp

Describe a 2-dimensional Gaussian slip distribution on the fault

- Flt# - fault number
- Srce# - source number for this fault
- Long, lat - longitude and latitude of center of source (in degrees)
- X-dim, W-dim - source dimensions along strike and down-dip (in kms)
- Amp - peak slip amplitude of source in mm

```
es: 1 1 167.0 -45.2 40 30 500
```

The source is described in fault plane coordinates x (along strike) and w (downdip). This option is currently is still in development, so use with caution and keep an eye on the results.

FA - fault information

FA: Fault_Name Fault_Number

[multiline data section]

Fault_Name = fault name (up to 10-characters, no spaces)

Fault_Number = fault number

The fault number has to be unique for each fault but not necessarily sequential in the file.

Nodes are placed along contours of the fault and numbered along strike, starting at the surface. Strike (positive X) is the direction faced if the fault dips to the right.

Multiline data section:

- Number of nodes along strike (max = MAX_x),
- Number of nodes downdip (max = MAX_z),
- Hangingwall block name (4-char name),
- Footwall block name (4-char name),
- Fault slip mode (0=shear only, 1-3D slip, see below)
- Downdip constraint (1=force downdip decrease in phi)
- Smoothing factor

for each depth:

- Depth of node in km,

for each node:

- Longitude of node degrees E,
- Latitude of node degrees N,
- {optional settings used for first depth only} Set azimuth of nodes going downdip (if using DD: or ZD: option) or add a new node.

```
fa: My_fault 2
  3 2 NaAm Paci 0 0 0
0
125 35
125 36
125 40
12
125.01 35.0
125.01 36.0
125.01 40.0
```

In the case above the fault strikes North (nodes input in order of increasing latitude), so the dip will be to the East.

- *Fault slip mode* - if set to 0, only shear is used on the fault (ie, Okada's U1 and U2), if set to 1, the U3 (tensional) component is used also. If this is changed, the Green's functions must be re-calculated.

- *Downdip constraint* - if set to 1, phi is forced to decrease downdip on fault (use '+ddc' in FL: option to apply to all faults).
- *Smoothing factor* - sets the maximum allowed along-strike variation in *phi*. The number given is the maximum allowable change in *phi* over one degree (111 km) of distance along strike. (Not applied if this value is zero.) This can be overridden by the **SM:** option.

Automated node generation at depth (DD: and ZD:).

Subsurface nodes defining a fault surface can also be set up automatically by the program. In this case you specify the surface nodes and the depth and dip angle to the nodes at depth using **DD:** or **ZD:**. For example:

```
fa:  SAF 2
    3 2 NOAM PACI 0 0 0
0.0
    125.  35.
    125.  36.
    125.  40.
dd:  6. 85.
dd:  8. 87.
```

The **DD:** option is followed by the incremental depth and dip to the next set of nodes. The **ZD:** option is the same except the depth given is the actual depth (not an incremental depth). The dip azimuth is taken as the normal to the fault strike. In the example above, the 2nd set of nodes will be 6 km deeper than the surface nodes and at a dip angle of 85° from them. The 3rd set of nodes will be 8 km deeper (at 14 km depth) than the second set and at a dip angle of 87° from them. An equivalent setup, using **ZD:** would be:

```
fa:  SAF 2
    3 3 NOAM PACI 0 0 0
0.0
    125  35
    125  36
    125  40
zd:  6. 85.
zd: 14. 87.
```

Changing fault dip azimuth. In the cases above, defnode determines the dip azimuth from the surface nodes to those at depth by taking the normal to the fault azimuth (i.e., dip azimuth = fault azimuth + 90°). The dip azimuth can be specified explicitly by entering a 1 as the third entry on the Lon, Lat line of the surface nodes followed by the desired dip azimuth. The example above would default to a dip azimuth of 90° since the fault strikes North. To change this to 95°, do:

```
fa:  SAF 2
    3 3 NOAM PACI 0 0 0
0.0
    125.  35.  1 95.
    125.  36.  1 95.
    125.  40.  1 95.
zd:  6. 85.
zd: 14. 87.
```

Adding along strike nodes. To add new nodes along strike, use a 2 as the third entry on the Lon, Lat line of the surface nodes. This will insert a new surface node halfway between this node and the next node. If you use **ZD:** or **DD:** the subsurface nodes will be built. Note that you are

increasing the number of along-strike nodes so you must change the relevant node assignment lines by hand (NN:, NV:, PN:, PV:, etc.).

```
fa: SAF 2
   3 3 NOAM PACI 0 0 0
0.0
 125. 35. 2
 125. 36. 2
 125. 40.
zd: 6. 85.
zd: 14. 87.
```

Changing fault dip along strike. Sometimes the dip angle of a fault may change along strike. This can also be accommodated automatically with the **DD:** or **ZD:** lines. In the **DD:** or **ZD:** line you specify first the depth increment (or depth for ZD:), then the dip angle, then (optionally) the number of nodes along strike that have this dip angle, then a new dip angle followed by the number of nodes with that dip angle, and so on. For example,

```
fa: SAF 2
   5 3 NOAM PACI 0 0 0
0.0
 125. 35. 1 95.
 125. 36. 1 90.
 125. 37. 1 90.
 125. 38. 1 90.
 125. 40. 1 95.
zd: 6. 85. 2 88. 3
zd: 14. 87. 2 89. 3
```

In this case, downdip from the first 2 surface nodes, the fault will dip at 85° to 6 km depth and at 87° to 14km depth. Downdip from the last 3 nodes, the fault will dip at 88° to 6 km depth and at 89° to 14km depth. The general format is:

```
zd: Z Dip1 N1 Dip2 N2 Dip3 N3 ...
```

where Z is the depth (km), Dip1 is the dip angle from the depth above to Z, N1 is the number of nodes along strike with this dip, N2 is the number of nodes along strike with dip angle of Dip2, and so on. The dip angles specified are always for the depth increment above the depth of Z. Make sure the sum of the N's equals the number of nodes in the along strike direction.

FF - add or remove selected faults

List the faults by number; negative to remove fault from inversion, positive to include in inversion. Default is all faults included in inversion.

```
ff: -5 +7
```

FL - set flags

```
FL: +ddc -cov +cos
```

The + flags turn the option on, the - flag turns it off. Default values are listed in parentheses.

```
+cos, -cos = do coseismic inversion (NO)
+cov, -cov = calculate parameter uncertainties (YES)
+cr2, -cr2 = use CRUST2 rigidities for calculating moments (NO)
+ddc, -ddc = force node phi values to decrease downdip on all faults (NO)
+dgt, -dgt = calculate residual strains and rotations at end (NO)
+eqk, -eqk = read earthquake file(s) and put in profile files (NO)
+for, -for = do forward model at end of run (YES)
+gcv, -gcv = use GPS NE covariances is estimating chi**2 (YES)
```


+inf, -inf = write MODL_info.out file (details of individual fault patches) (NO)
+inp, -inp = write each line of input file to screen to check reading errors (NO)
+mif, -mif = output blocks and faults to MapInfo .mif/.mid files (NO)
+pen, -pen = write penalties on iteration screen (NO)
+ph0, -ph0 = set phi for all faults to zero (remove all coupling) (NO)
+ph1, -ph1 = set phi for all faults to one (complete coupling on all faults) (NO)
+phf, -phf = set phi for all faults to current value and don't adjust them (NO)
+pos, -pos = make all longitudes positive, 0 to 360 degrees (YES)
+prm, -prm = use PREM rigidities for calculating moments (NO)
+rnd, -rnd = add random noise to output .vec file (NO)
+sim, -sim = write simplex in MODL_sa.out file (NO)
+tab, -tab = write table (.table file) of GPS velocities (NO)
+vtw, -vtw = read volcano file (votw.gmt) and put in profile files (NO)
+wcv, -wcv = write covariance matrix to file (NO)
+wdr, -wdr = write derivative matrix to file (NO)

Flags can be on multiple lines and more than one flag allowed per line. (Flags from previous versions are still supported.)

FS - calculate relative vectors and write to file

FS: filename BLK1 BLK2

Calculate the velocities of block BLK2 relative to block BLK1 at the lon, lat points contained in file 'filename'. Velocities are output in GMT psvelo format in file MODL_fslip.out. There can be several FS: lines.

Alternatively, or in addition, use

fsp: BLK1 BLK2 longitude latitude

to list points directly in the input file. For example,

fsp: NOAM PACI 241 40.8

fsp: NOAM PACI 241 39.8

FT - fault node parameterization type

ft: Fault #, Type

Parameterization of slip distributions on faults can be done in a number of ways. These fall into 3 classes: independent node values, node-depth-profiles with a 1-D distribution, and 2D distributions. In the independent node methods (Types 0 and 1), the node values can be independent in both strike and depth. In the node-depth-profile mode (Types 2, 3 and 4), slip at the nodes are prescribed functions of depth. In the 2D distribution (Type 6), node slip values are a prescribed Gaussian function of strike and depth.

Types of parameterization for the fault nodes:

Type = 0 independent nodes (each node can be a free parameter or nodes can be grouped)

= 1 independent nodes, phi decreasing down-dip (equivalent to type = 0, flag=+ddc)

(each node can be a free parameter)

constraint is $\phi(z+1) \leq \phi(z)$

= 2 modified Wang et al. (2003) function for $\phi(z)$; free parameters G, Z1, Z2 (Z2 > Z1)

G' = G (0.0 <= G <= 10.0)

G' = 20.0 - G (10.0 < G <= 20.0)

$\phi(z) = 1.0$ (z <= Z1)

$\phi(z) = \{ \exp [-(z'/G')] - \exp [-(1.0/G')] \} / \{ 1.0 - \exp [-(1.0/G')] \}$

for (Z1 < z < Z2)

where $z' = (z - Z1) / (Z2 - Z1)$

$\phi(z) = 0.0$ (z >= Z2)

G = shape parameter, Z1 = top of transition zone, Z2 = bottom of transition zone

= 3 boxcar $\phi(z)$; free parameters A, Z1, Z2 (Z2 > Z1)

$\phi(z) = 0$ (z < Z1)

$\phi(z) = A$ (Z1 <= z <= Z2)

$\phi(z) = 0$ (z > Z2)

A = Amplitude of boxcar, Z1 = upper depth, Z2 = lower depth

= 4 Gaussian $\phi(z)$; free parameters A, Zm, Zs

$\phi(z) = A \exp \{ -0.5 * [(z-Zm) / Zs] **2 \}$

A = peak amplitude, Zm = mean depth of distribution, Zs = spread of distribution

= 5 not used

= 6 2D Gaussian $\phi(x,w)$; free parameters Lon, Lat, A, sX, sW,

x is along strike dimension, w is down dip dimension on fault (not depth)

$\phi(x,w)$ is specified by Lon, Lat of center, peak amplitude A, distance scales in X and W

(see ES option)

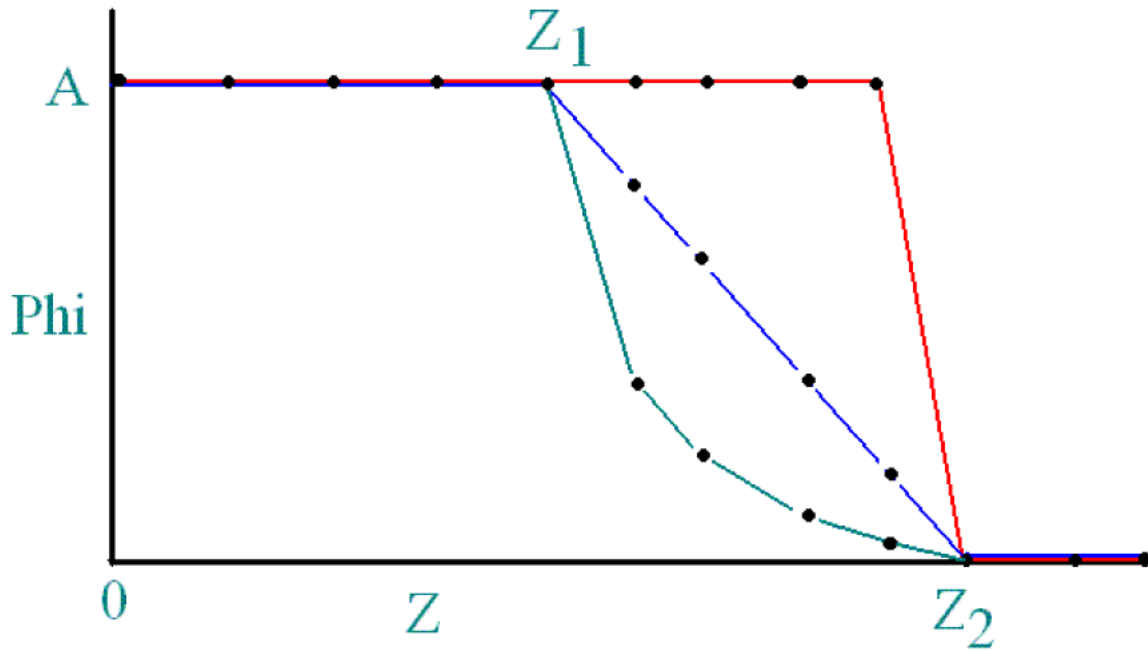
$\phi(x,w) = A * \exp(-0.5 * (dX/sX) **2) * \exp(-0.5 * (dW/sW)**2)$

A = amplitude; dX, dW = source to node offsets in X and W;

sX and sW = source spread in X and W

ft: 2 4

Controls for types 0 and 1 are through options NN:, NV: and NX:. For types 2 - 4 use PN:, PV:, and PX:.



Node z-profile types.

Some Examples:

1. All nodes have independent phi values and are all free parameters.
2. # set fault 2 to type 0 (free nodes)
3. FT: 2 0
- 4.
5. # fault 2 has 6 nodes along strike, 3 downdip. Each node has a unique index number.
6. NNg: 2 6 3
7. 1 2 3 4 5 6
8. 7 8 9 10 11 12
9. 13 14 15 16 17 18
- 10.
11. # an equivalent NN: line is
12. NN: 2 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
- 13.
14. # initial phi values
15. NVg: 2 6 3
16. 1.0 1.0 1.0 1.0 1.0 1.0
17. 0.7 0.7 0.6 0.6 0.5 0.5
18. 0.3 0.3 0.3 0.3 0.3 0.3
- 19.
20. ## or
21. NV: 2 1.0 1.0 1.0 1.0 1.0 1.0 0.7 0.7 0.6 0.6 0.5 0.5 0.3 0.3 0.3
0.3 0.3 0.3
- 22.
23. # to force the phi values to decrease downdip, use either
24. FT: 2 1
25. # or to apply to all faults
26. FFlag: +ddc

```

27.
28. # to fix the 6 surface nodes at phi = 1, use
29. NX: 2 1 2 3 4 5 6
30.
31. # or, more easily make all the surface nodes have the same index and
    fix that one index
32.
33. NNg: 2 6 3
34. 1 1 1 1 1 1
35. 2 3 4 5 6 7
36. 8 9 10 11 12 13
37. NV: 2 1.0 0.7 0.7 0.6 0.6 0.5 0.5 0.3 0.3 0.3 0.3 0.3 0.3
38. NX: 2 1
39. Full coupling from surface to depth  $Z_2$ , no coupling below  $Z_2$ , let  $Z_2$  vary along strike
    (red curve above). Use boxcar option, fix  $A = 1$ , fix  $Z_1 = 0$ , solve for  $Z_2$ .
40. # set fault 1 to type 3 (boxcar)
41. FT: 1 3
42.
43. # fault has 6 nodes along strike, they will all be different
44. PN: 1 1 2 3 4 5 6
45.
46. # fix the first and second parameters (A and Z1)
47. PX: 1 1 2
48.
49. #  $A = 1$  and  $Z_1 = 0$  for all profiles,  $Z_2$  is variable and will be
    adjusted
50. PV: 1 6
51. 1 1 1 1 1 1
52. 0 0 0 0 0 0
53. 30 35 40 30 35 40
54.
55. Full coupling from surface to depth  $Z_1$ , linear decrease to depth  $Z_2$  and no coupling below
     $Z_2$ . Fix  $Z_1$  at 10 km, let  $Z_2$  vary along strike (blue curve above). Use Wang option, fix  $A$ 
    = 10, fix  $Z_1 = 10$ , solve for  $Z_2$ .
56. # set fault 1 to type 2 (Wang)
57. FT: 1 2
58.
59. # fault has 6 nodes along strike, they will all be different
60. PN: 1 1 2 3 4 5 6
61.
62. # fix the first and second parameters (A and Z1)
63. PX: 1 1 2
64.
65. #  $A = 10$  and  $Z_1 = 10$  for all profiles,  $Z_2$  is variable and will be
    adjusted
66. PV: 1 6
67. 10 10 10 10 10 10
68. 10 10 10 10 10 10
69. 30 35 40 30 35 40
70.
71. Full coupling from surface to depth  $Z_1$ , exponential decrease (Wang) to depth  $Z_2$  and no
    coupling below  $Z_2$ . Let  $Z_1$ , let  $Z_2$ , and  $A$  be adjusted and vary along strike (green curve
    above).
72. # set fault 1 to type 2 (Wang)
73. FT: 1 2

```

```
74.
75. # fault has 6 nodes along strike, first 2 are the same, rest are
    different
76. PN: 1 1 1 2 3 4 5
77.
78. # no PX line as all parameters are free
79.
80. # Starting A = 10, Z1 = 5, Z2 = 30 and all will be adjusted
81. # second argument of PV line is now 5 as there are now only 5 unique
    sets of
82. # parameters, per the PN: line
83. PV: 1 5
84. 10 10 10 10 10
85. 5 5 5 5 5
86. 30 30 30 30 30
```

FX - fix node position

fx: Fault #, Node X-index, Node Z-index, Longitude, Latitude

Force node given by Fault #, Node X-index, Node Z-index to be at Longitude and Latitude.

Overrides all other position specifications (ie, implemented after FA: and MV: lines).

fx: 2 10 5 120.3 32.3

The 10th X node and 5th Z node of fault 2 will be at long=120.3, lat=32.3

GD - Green's function directory, step sizes and parameters

GD: Directory X_step W_step Flag dx_node x_near x_far

Generate Green's functions. GFs are calculated for all faults as needed.

- 'Directory' is for Green's functions files. Default is 'gfs'. Directory name must be 3 characters, no spaces.
- X_step, W_step are sizes of patches along fault surfaces for integration between nodes (for GFs only). X_step - length of fault patch (in km) along strike (default 10 km); W_step - length of fault patch (in km) downdip (default 5 km). The X, W interpolation values specified here can be different from those in the IN: option.
- Flag = 0, generate GFs as needed, = 1 force new GFs for all faults
- dx_node = tolerance (in km) of new node position that triggers calculation of new Green's functions (default 1 km)
- x_near = proximity (in km) of sites that use same Green's functions (default 1 km)
- x_far = don't generate GFs for sites farther than x_far (in km) from any node (default 1500 km)

gd: gf1 2 1 0 1 1 1000.

will place GFs in directory 'gf1'. Step size for GF integration is 2 km along strike, 1 km downdip.

Before generating a new GF, the program checks whether or not the current GF is up to date by looking at the node position, surrounding node positions, the interpolation distances (if the new ones are greater than or equal to the stored ones, new GF is not generated). If the stored GF does not match, a new one is generated. Sometimes, this checking can fail, for example if you add new data. To override the checking and regenerate all GFs, set Flag = 1 (or manually delete all the GF files).

gd: ggg 5 2 1
will force generation of all new GFs.

The GFs are in ASCII files named gf001001001g, gf001002001g, etc. in the directory specified by the GD: option. (File names; first 3 digits are fault number, second 3 are the along strike node index, the third 3 are the downdip node index, and the final letter designates the data type; g - GPS, u - Uplift, t - Tilts, s - strain rates)

Important note: If you add new data you must re-generate all GFs. The program cannot append new GFs to the existing GF files.

GI - rotate GPS velocity fields into reference frame

gi: N1 N2

List of GPS velocity field poles to adjust during the inversion. The listed poles correspond to the pole index number given in the GP: line. This applies a 3-parameter rotation to the velocity field in the inversion to fit the reference frame better.

gi: 2 4

Adjusts velocity field poles 2 and 4.

The GPS sites in a particular field should not be on a single block and should have some overlap with other GPS solutions or the reference frame block.

GP - GPS file

GP: NAME filename N { F Wx Wy Wz Smin Smax RSmax }

Read GPS data file

- NAME = 4-char code name for this GPS velocity file
- filename = file containing data
- N = index for the GPS velocity field rotation pole (used in GI: option)
- F = sigma scaling factor (each sigma multiplied by F so weight is multiplied by $1/F^{**2}$; default = 1.0)
- Wx Wy Wz are the components of the angular velocity vector that puts these vectors into the desired reference frame.
- Smin = minimum velocity sigma for this file (if sigma is less than Smin, sigma = Smin)
- Smax = maximum velocity sigma for this file (if sigma is greater than Smax, velocity is not used)
- RSmax = if Residual/Sigma exceeds RSmax, velocity is not used (use this with caution, if at all)
- (if Smin, Smax, RSmax are zero, these are not applied)

```
gps file: IND1 "../data/indo1.vec" 1 2.0 -.12 .20 1.22
```

File format (use GMT psvelo -Se format)

```
lon lat Ve Vn SigVe SigVn NE_Corr Site_name
```

Site_names are stored as 8-character strings.

WARNING: If a site name starts with a number, defnode may choke on the file while trying to read in free format. In this case, you can format the input file and include a format line at the top of the file. The program looks for an open parentheses symbol in the first column to indicate a format line. For example:

```
(2f8.3, 4f8.1, f8.4, 1x, a8)
243.111 35.425 -19.4 -6.1 0.6 0.4 0.0014 001D
240.375 49.323 -13.1 -11.8 0.6 0.4 0.0018 4750
212.501 64.978 -8.1 -22.3 0.6 0.4 0.0036 47SB
243.411 35.825 -14.4 -4.1 1.6 1.4 0.0014 0001_edm
```

Another way is to put the site names in quotes, ie "001D".

You can read in multiple GPS velocity files up to MAX_gpsfiles.

GR - grid

GR: N X_start Number_of_X_steps X_step Y_start Number_of_Y_steps Y_step
Surface grid - calculations made at points in a regular grid. Output files MODL_grid_N.info and MODL_grd_N.vec (see format below) can be countoured with GMT's pscontour or plotted with psvelo. Can now output multiple grids. N in file names is grid number (up to MAX_grids).

- Grid index
- Starting X longitude
- Number of X steps
- X step in degrees
- Starting Y latitude
- Number of Y steps
- Y step in degrees

```
gr: 1 245.1 40 0.1 23.1 50 0.1
```

GS - grid search controls

GS: #grid_steps parameter_grid_step #grid_searches search_type
Set controls for parameter grid search.

- #grid_steps - This integer controls the number of grid steps (max = 100).
- grid_step - step for searching parameter values.
- #grid_searches - number of times the grid search goes through all the parameters.
- search_type -
 - 0 - search the full range of parameters in order
 - 1 - search full range in random parameter order
 - 2 - gradient search (follow gradient from current parameter value)
 - 3 - gradient search in random parameter order

If $N = \#grid_steps$, S is the $grid_step$ and P is the current best value of the parameter, the parameter will be searched from $P-N*S$ to $P+N*S$ in steps of S . For each grid search, this step value S is decreased. If gradient search is being used, it will step down the gradient in the χ^2 until it reaches a minimum.

Grid search is run before simulated annealing.

```
gs: 10 0.1 3 0
```

HC - hard constraints

```
hc: I Lon Lat BLK1 BLK2 Lower_value Upper_value
```

Hard constraints - force value of slip rate or slip azimuth on fault to fall in specified range

I = 1 for slip rate constraint

I = 2 for slip direction constraint

I = 3 for rotation rate constraint

BLK1 = moving block

BLK2 = fixed block

This works by applying a severe penalty for values falling outside the specified range.

Results are put in MODL.hcs file

```
hc: 1 Noam Paci 232.5 32.1 24.0 34.0
hc: 2 Noam Paci 232.5 32.1 280.0 320.0
```

IN - fault interpolation step sizes

```
IN: dX dW
```

Sizes of patches along fault surfaces for integration between nodes (for the forward solution and plot file only). dX - length of fault patch (in km) along strike, dW - length of fault patch (in km) downdip direction. In general these should be the same as in the $GF:$ option. To speed up preliminary runs these can be made larger than in the $GF:$ option. These interpolation values are used for the grid ($GR:$) and profile ($PR:$) calculations as well as for the plot file $MODL_flt_atr.gmt$. To really speed things up if you want to make the plot files only (without calculations) use the flag $-for$ (no forward calculations).

```
in: 4 4
```

MF - merge faults

```
MF: M N
```

Merge faults M and N at a T-junction. Fault M is truncated against fault N . The truncated end of fault M follows the plane of fault N downdip.

```
mf: 1 3
```

```
          3
          3
1 1 1 1 1 1 3
          3
          3
          3
```


This is not always succesful - you can use the FX: option to force nodes to be where you want them.

MM - min and max moment for fault

MM: N M1 M2

The seismic moment for fault N is constrained to fall between M1 and M2 (in N-m). This can be used to damp slip models.

mm: 3 0.0 1.2e20

MO - model name

MO: MODL

Model name - 4 characters, used as prefix to name output files and directory. A directory with this name will be created and all output files placed in it.

model: indo

See also EM:

MV - move surface points

mv: x1 y1 x2 y2

Move all occurrences of point x1, y1 to x2, y2. Applied to block boundaries and faults.

mv: 120.21 43.21 120.25 43.22

NI - number of iterations

ni: N

Run both the simulated annealing and grid search N times.

ni: 2

NN (NF) - node indices

NN: F I I I I I I I

Node indices

F = fault number

I = parameter index, one for each node on fault, in order (see introductory notes for ordering of nodes).

Each node on the fault is assigned an index number which is used to assign properties to it (its phi, inversion characteristics).

If the index is not zero or in the fixed node list, the node is a free parameter in the inversion.

The initial slip ratio (phi) for this node is taken from the list of node phi values (the NV: input line). For example, if the node has index 5 assigned, it is assigned the phi that is fifth in the NV: list for this fault.

In the example below, the first 3 nodes of fault 1 have slip ratio (phi) values of 0.1, the next 3 have phi values of 0.2, the next 3 nodes have phi = 0.3, and the last 3 are zero. The nx: line fixes the last 3 nodes at phi=0 in the inversion.

```
nn: 1 1 1 1 2 2 2 3 3 3 4 4 4
nv: 1 .1 .2 .3 0.
nx: 1 4
```

For multiple faults, the index numbering starts with 1 for each fault:

```
nn: 1 1 1 1 2 2 2 3 3 3 4 4 4
nv: 1 .1 .2 .3 0.
nx: 1 4

nn: 2 1 1 2 2 3 3
nv: 2 .3 .6 .9
```

An alternative, more intuitive input format is available by using NNg: In this case the node indices are entered in a grid, mimicking their spatial relation on the fault.

```
NNg: 3 4 5
     1 1 2 2
     3 3 4 4
     5 5 5 5
     5 5 5 5
     0 0 0 0
```

The first argument after the NNg: is the fault number, then the number of nodes along strike, then number of nodes down dip. The node indices are then listed in a grid fashion.

The example above is equivalent to the single line NN: form:

```
nn: 3 1 1 2 2 3 3 4 4 5 5 5 5 5 5 5 5 0 0 0 0
```

NV (NO) - node phi values

NV: F V1 V2 V3 V4 V5

Node slip ratio values or coseismic slip amounts (mm)

F = fault number

V = Slip ratio (phi) or slip (mm) values for node indices. For example, any node that is assigned a 1 in the NN: line will be assigned phi = V1. This line should contain the number of phi values equal to the number of different indices in the NN: line.

```
nv: 1 .6 .4 .3
```

Alternatively, NV:, like NN:, has a grid form implemented with a 'g' in the third column. The lines:

```
nn: 1 1 1 1 2 2 2 3 3 3 4 4 4
nv: 1 .1 .2 .3 0.
```

can be equivalently entered as:

```
nng: 1 3 4
     1 1 1
     2 2 2
     3 3 3
     4 4 4
```

```
nvg: 1 3 4
```

```
.1 .1 .1  
.2 .2 .2  
.3 .3 .3  
0. 0. 0.
```

If NV: not specified, phi values are assigned as a decreasing function of depth.

NX - fix node value

nx: F I I I I I

Specifies which nodes are to be fixed (ie, not a free parameter) in the inversion

F = fault number

I = node index to be fixed

nx: 5 2 3

will fix any nodes with indices of 2 or 3 in fault 5

PE - set penalty factors

PE: N Factor

Penalty factors for constraints. When a parameter value strays outside the allowed range, this factor is multiplied by the difference and added to the penalty. See PM: option for setting parameter ranges.

1 - Moment
2 - Node values
3 - Depths
4 - Dwindip constraints
5 - Smoothing along strike
6 - Hard constraints

pe: 3 50

Set penalty factor for depths to 50.

PF - parameter file

PF: filename N

Specify a filename to hold the model parameter values. The number N controls reading/writing of the parameters.

- N=1 read parameters from the file,
- N=2 write parameters to file,
- N=3 read and write parameters from/to file.

Reads take place prior to inversion, writes take place after inversion.

pf: bestfit.io 3

Temporary parameter files will also be generated. A final file with the name MMMM_pio.YYYYMMDD will be generated (YYYY=year, MM=month, DD=day).

PG - pole for GPS file

PG: NAME Latitude Longitude Omega

or

PGc: NAME Wx Wy Wz

Pole of rotation for GPS file to put it in reference frame

NAME = GPS file short name (4-char) from GP: line, latitude, longitude, omega are pole; OR
(Wx, Wy, Wz) are Cartesian components of angular velocity vector in deg/Ma

pg: INDO -12.0 123.0 0.2

pgc: PNW1 .1 -.3 .8

The 'c' indicates pole is in Cartesian coordinates.

PI - block poles to adjust

PI: N N N

List the block poles to adjust in the inversion

pi: 2 5

adjust poles 2 and 5 in the inversion, keep all other poles fixed. (Note that this does not necessarily mean blocks 2 and 5, since poles 2 and 5 may be assigned to other blocks.)

Use the GI: option to adjust the poles of GPS velocity fields.

PM - parameter min and max values

PM: N Min_value Max_value

Set the minimum and maximum limits on parameter types. Parameter N is constrained to fall between Min_value and Max_value.

N - parameter type

-
- 1 - GPS velocity field pole component (deg/Ma)
 - 2 - block pole component (deg/Ma)
 - 3 - strain rate component (nanostrain/yr)
 - 4 - Slip amplitude (mm)
 - 5 - Modified Wang gamma value (dimensionless)
 - 6 - minimum locking depth (kms)
 - 7 - maximum locking depth (kms)
 - 8 - mean locking depth for 1D Gaussian $\phi(z)$ (kms)
 - 9 - spread of locking depth for 1D Gaussian $\phi(z)$ (kms)
 - 10 - Longitude for 2D Gaussian $\phi(x,z)$ (degrees)
 - 11 - Latitude for 2D Gaussian $\phi(x,z)$ (degrees)
 - 12 - Along strike spread for 2D Gaussian $\phi(x,z)$ (kms)
 - 13 - Downdip spread for 2D Gaussian $\phi(x,z)$ (kms)
 - 14 - not used
-

Set the minimum locking depth to be between 0 and 5 kilometers.

pm: 6 0 5

PN - node-depth-profile indices

PN: F I I I I I

F = fault number

I = node-depth-profile indices

The nodes on faults can be parameterized as specified functions of depth. In this mode, each node-depth-profile starts at the surface node and goes downdip along the fault (each node in the node-depth-profile therefore has the same x-index). A fault has the number of node-depth-profiles equal to its number of surface points. The phi values in a node-depth-profile follow a specified function of depth (see FT: option).

The node-depth-profile parameters are controlled by the PN:, PV:, and PX: options much like the NN:, NV:, and NX: options control the individual node parameters.

```
PN: 1 1 1 2 2 3 4 4
```

In this example, fault 1 has 7 node-depth-profiles along strike. The first 2 will have the same parameters, the 3rd and 4th will have equal parameters, and so on.

Example below is a boxcar function of depth for fault 1. The first 3 of the 6 node-depth-profiles have same parameters and second set of 3 have same parameters. The PV: option gives the initial parameter values for the indices. Node-depth-profiles 1 to 3 are assigned index 1 in the PN: statement and node-depth-profiles 4-6 have index 2. Index 1 has parameters of 0.5, 10 and 30 so these are assigned to node-depth-profiles 1 to 3. In the inversion node-depth-profiles 1 to 3 will always have the same parameters since they are assigned the same index.

```
FT: 1 3
PN: 1 1 1 1 2 2 2
PV: 1 2
    .5 .8
    10 10
    30 25
```

PO - angular velocities (poles) for blocks

po: N Lat Lon Omega

or

poc: N Wx Wy Wz [covariance ellipse]

or

pob: Block_name Wx Wy Wz [covariance ellipse]

or

pof: N Wx Wy Wz [covariance ellipse]

Poles of rotation

N = pole number, then lat, lon, and omega (deg/Ma) of pole

```
po: 1 0 0 0 (use for reference frame block)
```

```
po: 2 -10 145 -.37
```

```
po: 3 45 245 1.3
```

3rd char = c for Cartesian pole specification

= f to overwrite parameter file value (Cartesian only)

= b to use block name to specify pole (Cartesian only)

Cartesian: enter Wx, Wy, Wz in degrees/Ma

then Sx, Sy, Sz standard errors in deg/Ma

then Sxy, Sxz, Syz the unitless covariances

```
poc: 5 -1.2 0.4 1.1 0.12 0.23 0.12 0.001 0.002 0.112
```

```
pob: Paci -1.2 0.4 1.1 0.12 0.23 0.12 0.001 0.002 0.112
```

If you're using the PF: option (parameter file), use POf: to fix a pole at a specified value in the inversion (use Cartesian angular velocity representation and remove pole number from PI: list). This pole vector then overrides the pole values read in from the parameter file. Alternatively you can edit the parameter file and change the pole values.

```
pof: 5 -1.2 0.4 1.1 0.12 0.23 0.12 0.001 0.002 0.112
```

PR - profile

```
pr: N Xo Yo M dX Az Hwdth
```

Creates GMT plottable files to generate profile lines.

- N - Line number
- Xo - starting Longitude (degrees E)
- Yo - starting Latitude (degrees N)
- M - number of points
- dX - distance step (degrees)
- Az - azimuth of profile (degrees clockwise from North)
- Hwdth - half-width (km) for plotting observations along profile line

```
pr: 1 245 35 50 .05 0 25  
profile 2, 45 degrees: 2 126.5 -4. 30 .05 45 60
```

PV - node-depth-profile parameter values

Sets the initial values of the parameters for fault types 2 to 4.

```
PV: F N  
A A A A  
Z1 Z1 Z1 Z1  
Z2 Z2 Z2 Z2
```

F = fault number, N = number of columns of parameter values to follow.

```
PV: 3 4  
10 8 3 10  
10 10 10 10  
30 30 30 35
```

PX - node-depth-profile fixed parameters

Specify which parameters are fixed for fault types 2 to 4.

```
PX: F I I
```

F = fault number, I = parameter number to fix

See FT: for the parameters for each fault type.

```
FT: 2 3  
PX: 2 1 3
```

Fault 2 is boxcar (type 3), and the amplitude (parameter 1 for boxcar) and bottom depth (parameter 3 for boxcar) are fixed.

RC - remove GPS vectors from circular region

RC: Lon Lat Radius

Remove selected GPS sites within Radius (in km) from point at Lon, Lat

rc: 143.2 43.4 20.0

RE - reference frame

RE: Block_name

Block Block_name is reference frame for vectors. If GPS vectors are not in this reference frame, use the GI option to find the rotation to put them in the reference frame.

reference block: NOAM

You can set the reference frame to something other than a block (eg NNR or ITRF) by making a fictitious block and setting it to be the reference frame.

RF - output vectors in new reference frame

RF: Wx Wy Wz

(Wx, Wy, Wz) gives angular velocity of rotation to apply. The velocities from this pole are subtracted from the model velocities. Output velocities in this new reference frame will be in MODL_newf.obs (observed velocities) and MODL_newf.vec (calculated velocities).

RM - remove selected GPS vectors

RM: GPS_name site1 site2 site3

or

RMb: Blk1 Blk2 Blk3

or

RM8: sit2_GPS sit3_GPS

remove selected GPS sites

rm: SCEC GOLD SPN1 AREQ

rm: PNW1 HOB1 YBHB

rm: **** FAIR

rmb: NoAm Paci

The first entry is the 4-char name of the velocity solution (defined in GP: option). The site names that follow will be removed if they are in that velocity solution. Use **** to remove the sites from ALL solutions (for example, above FAIR will be removed from all solutions). Up to 50 entries per line, multiple lines allowed (up to MAX_rm).

To specify 8-character GPS site names, put an '8' in 3rd column:

RM8: SCEC GOLD_GPS site_EDM

To remove all GPS sites from a particular block, put a 'b' in 3rd column:

RMb: NoAm Paci

where NoAm and Paci are block names.

RO - rotation rate data

RO: filename F

Rotation rate data file

F = weight factor (F is multiplied by all sigmas)

Format of data file:

```
Long Lat Rot_Rate Sigma Identifier
```

Rates in deg/Ma, clockwise is negative, Identifier is 40-char

```
ro: "../data/rot.dat" 3
```

Alternative is to put all on one line, use 'd' in 3rd column

```
rod: Long Lat Rot_Rate Sigma Identifier
rod: 243.2 25.3 -0.7 0.3 "So_and_so paleomag"
```

RS - reference site for GPS vectors

```
RS: N SITE
```

GPS vector field N is relative to site SITE. The calculated vector at SITE is subtracted from all others in field N.

SA - simulated annealing inversion controls

```
SA: T I A1 A2
```

Run simulated annealing and sets controls

- T = temperature (set to 0 for downhill simplex)
- I = number of iterations
- A1, A2 = min and max allowed values for node phi values (0 and 1 for interseismic)

```
sa: 100 250 0 1
```

Without the GS: or SA: line, the program will do forward model only if +for flag is set and will make plot files only if -for flag set.

SI - strain rate tensors to adjust

```
SI: N N N
```

List the strain rate tensors to adjust in the inversion

```
si: 2 5
```

adjust tensors 2 and 5 in the inversion, keep all others fixed. (Note that this does not necessarily mean blocks 2 and 5, since tensors 2 and 5 may be assigned to other blocks.)

Strain rate tensors are calculated for a spherical Earth using the formulas in Savage et al. (2001).

SK - skip input

SKip:

Skip over following input lines until **CO:** (continue) line is found. Allows skipping many lines of input data.

SM - along-strike smoothing of fault coupling

SM: Fault_number smoothing_factor

Smoothing factor - sets the maximum allowed along-strike variation in *phi*. The smoothing_factor given is the maximum allowable change in *phi* over one degree (111 km) of distance along strike. (Not applied if this value is zero.)

sm: 5 0.4

SN - snap polygon points together

SN: tolerance_in_km

Force points along adjacent block polygons to have the same values, to remove small gaps and overlaps. If the points are within tolerance_in_km of each other in distance, they are both assigned a value equal to their average.

sn: 5.0

SR - fault slip rate data

Three input options are available:

- From a file with blocks specified in control file:

```
sr: filename BLK1 BLK2 F Smin Type
```

format of file: Long Lat V1 V2 Azimuth

- From file with blocks specified in file:

```
srf: filename F Smin Type
```

format of file: BLK1 BLK2 Long Lat V1 V2 Azimuth Label

- Read as line directly from control file:

```
srd: BLK1 BLK2 Long Lat V1 V2 Azimuth Smin Type Label
```

where

- filename = slip rate (or spreading rate) data file
- Slip rates (in mm/yr) are between Block BLK1 and block BLK2.
- F = scaling factor (F multiplied by all sigmas)
- Azimuth = azimuth of rate measurement (ship track direction, for example). If Azimuth = 0, the total slip rate is used.
- Smin = minimum sigma allowed
- Label = label (40-chars, no blanks or put in quotes)
- Type

- if Type = 0, V1 = mean rate, V2 = rate sigma, treat as Gaussian data
- if Type = 1, V1 = mean rate, V2 = rate sigma, treat as Uniform (min/max) data; min = V1-V2, max=V1+V2; sigma = V2
- if Type = 2, V1 = min rate, V2 = max rate, treat as Uniform (min/max) data; sigma = (V2-V1)/4
- if Type = 3, V1 = min rate, V2 = max rate, treat as Gaussian data; mean = (V1+V2)/2; sigma = (V2-V1)/4 (assumes min/max range is +/- 2 sigma)

```
sr: saf_rate.dat NOAM PACI 1 0 0
```

where saf_rate.dat looks like

```
242.2 33.3 25.4 3.4 -320
```

or

```
srf: saf_rate.dat 1 0 0
```

where saf_rate.dat looks like be

```
NOAM PACI 242.2 33.3 25.4 3.4 -320
```

For Gaussian fitting the penalty is the $(\text{residual}/\text{sigma})^{**2}$, where the residual is the difference between the calculated value and the mean observed value. For the uniform fitting, the residual is how far the calculated value falls outside the min/max range of the observed value and the penalty is the $(\text{residual}/\text{sigma})^{**2}$.

SS - strain rate data

SS: filename F

Horizontal surface strain rate data file

F = scaling factor (F multiplied by all sigmas)

```
ss: strains.dat 2
```

Format of data file (strain rates in nanostrain/yr):

Two formats are allowed; in one the strain rate tensor is in the form of principal axes, the other is in N, E coordinate system

input lines of form:

```
Lon Lat Radius Type E1 sigE1 E2 sigE2 E3 sigE3 {Network Name}
```

Lon, Lat are of network centroid, Radius is approximate radius of network in kilometers,

if Type = 0 shear strain rates are read in E,N (x,y) coordinates; E1 = E_{xx} , E2 = E_{xy}, E3 = E_{yy}

if Type = 1 principle strain rates are read in and converted to E,N coordinate system; E1 = maximum strain rate (contraction is negative), E2 = minimum strain rate, E3 = Azimuth of maximum strain rate

ST - strain rate tensor

ST: I Exx Eyy Exy {Cx Cy}

For strain rate tensor I, values are given in nanostrain/year. Cx and Cy are the longitude and latitude of the origin for this strain rate tensor - these default to the centroid of the block but should be specified if the tensor is for multiple blocks.

st: 4 3.2 4.1 6.2 234.2 -43.2

SV - slip vector data

Slip vector or transform fault azimuth data

Read from file:

sv: filename BLK1 BLK2 F

Format of data file:

Long Lat Azimuth Sigma

Read from file:

svf: filename F

Format of data file:

BLK1 BLK2 Long Lat Azimuth Sigma

Read from within control file

svd: BLK1 BLK2 Long Lat Azimuth Sigma Label

Slip vector azimuth or transform fault azimuth is between block BLK1 and block BLK2. BLK1 is the fixed block, BLK2 is the moving block. BLK2 moves at the given azimuth relative to BLK1. Azimuths in degrees clockwise from North.

F = scaling factor (F multiplied by all sigmas)

Label is 40-char description (no spaces); for file formats label is filename.

sv: "../svs/slip_vec.dat" NOAM PACI 5

TI - tilt rate data

TI: filename F

Tilt rate data file

F = scaling factor (F multiplied by all sigmas)

tilt rate and sigma in nanoradians/year

ti: "data/tilt.dat" 1.0

Format of data file:

Lon1, Lat1, Lon2, Lat2, Tilt_rate, Tilt_rate_sigma, NAME(4-chars)

Lon1, Lat1 and Lon2, Lat2 are endpoints of profile over which the tilt rate is measured. Tilt_rate is positive if uplift increases along profile.

UP - uplift rate data

UP: filename F

Uplift rate data file

F = scaling factor (F multiplied by all sigmas)

up: /data/up.dat 1.0
uplift rates in mm/a, up is positive

Format of data file:

```
Lon Lat Uplift_rate Sigma {Site name}
```

use up1: if format is Lat Lon Uplift_rate Sigma

A fortran format can also be specified by placing it at the beginning of the input file starting in the first column. For example:

```
(4f8.1, 1x, a4)
243.111 35.425 -1.4 0.6 001D
240.375 49.323 -1.1 0.6 4750
212.501 64.978 2.1 0.6 47SB
```

ZD - specify dip and depth of fault segment

see FA: option

OUTPUT FILES

All output files are put in model directory.

Not all of these listings have been checked intensely for for accuracy, some could be different than shown. Let me know if something looks awry.

(MODL = 4-character experiment name, from MO: option, default = 'temp')

MODL.cov	- covariance matrix (needs '+wcv' flag set)
MODL.der	- derivatives matrix (needs '+wdr' flag set)
MODL.dgt	- residual strain rates and rotations for blocks (needs +dgt flag set)
MODL.hcs	- result of hard constraints
MODL.net*	- GPS network adjustment velocities
MODL.nod	- summary of node information
MODL.obs*	- observed GPS vectors with re-scaled uncertainties
MODL.omr*	- observed GPS vectors minus rotational part
MODL.pol	- summary of poles (relative poles for all block pairs)
MODL.prm	- input parameters
MODL.res*	- GPS vector residuals
MODL.rot*	- rotational part of predicted GPS velocity field

```

MODL.slp*           - fault strain deformation part of predicted GPS velocity
field
MODL.str*           - internal strain part of predicted GPS velocity field
MODL.srs            - summary of fits to slip rate data
MODL.sss            - summary of fits to strain rate data
MODL.svs            - summary of fits to slip vectors/fault azimuths
MODL.tlt            - summary of fits to tilt rate data
MODL.ups            - summary of fits to uplift rates
MODL.vec*           - predicted GPS vectors
MODL_blk.gmt        - plot file of block outlines
MODL_blk2.gmt       - plot file of block boundaries separating blocks with same
pole
MODL_blk3.gmt       - plot file of faults
MODL_blocks.out     - block information output (see below)
MODL_grid_N.vec     - predicted vectors for grid points, N is grid number (GR:
option)
MODL_fslip.out      - relative velocities at requested points (from FS: option)
MODL_info.out       - information on each subsegment of faults
MODL_lin.gmt        - coordinates of profile lines for putting on map
MODL_model.input    - poles, block boundaries and faults in control file format
MODL_control.backup - copy of control file
MODL_mid.vec        - relative block vectors on faults at midpoints between
surface nodes
MODL_obs.gmt        - tilt lines and strain network polygons for plotting
MODL_flt_atr.gmt    - plot file for fault parameters (see below)
MODL_pNN.out        - output for profile number NN (see below)
MODL_removed.vec*   - vectors removed from inversion
MODL_sa.out         - summary of simulated annealing run with final solution
MODL.summary        - summary of fits to data, poles, blocks
MODL.table          - comprehensive table of GPS velocities

```

Files ending with .gmt are plot files for GMT and are described in the PLOTTING section.

The **vector files** followed by an asterisk above are in GMT's psvelo -Se command format.

```

format: (2f10.4, 2f10.2, 2f8.2, f9.4, 2x, a8, 3(1x,a4))
Items:
1. Longitude
2. Latitude
3. East_velocity
4. North_velocity
5. East_sigma
6. North_sigma
7. NE_correlation
8. Site_name (8-characters)
9. GPS velocity field 4-char code
10. Block 4-char code
11. Pole 4-char code

```

After the site_name, each line contains the GPS velocity field name, the block name, and the pole number, each 4-characters, for example:

```

    245.5190  62.4810   -0.73   -0.12   0.17   0.17   0.0000  YELL_GPS
ITRF NoAm P005

```

where YELL_GPS is the site name, ITRF is the velocity field name, NoAm is the block name, and P005 is the rotation pole (i.e., the pole this vector constrains is #5 in the solution).

MODL_fslip.out - relative velocities between blocks at requested points (from FS: option). In psvelo -Se format. The site_name field has the block pairs and this is followed by the total velocity and its uncertainty.

```
format (2f10.4, 4f8.1, f9.4, 1x, a9, 2f8.1)
```

Items:

1. Longitude
2. Latitude
3. East_velocity
4. North_velocity
5. East_sigma
6. North_sigma
7. NE_correlation
8. Block pairs (9-characters)
9. Total velocity
10. Total velocity sigma

```
241.0000 35.5000 -11.8 7.6 0.3 0.2 0.0727 SNEV_NOAM  
14.1 0.2
```

MODL_mid.vec - fault information at mid-points between each pair of nodes along faults

```
format(2f8.3, 4f7.1, f7.3, 1x, a10, 2(1x,a4), i4, 4f6.1, 2f7.3, 6f6.1)
```

Items:

1. Longitude
2. Latitude
3. East_velocity
4. North_velocity
5. East_sigma
6. North_sigma
7. NE_correlation
8. Fault name
9. Hangingwall block
10. Footwall block
11. Fault number
12. Total velocity
13. Total velocity sigma
14. Velocity azimuth
15. Velocity azimuth sigma
16. Phi value
17. Phi value sigma
18. Velocity parallel to fault (positive is right-lateral)
19. Velocity parallel to fault sigma
20. Velocity normal to fault (positive is divergence)
21. Velocity normal to fault sigma
22. Fault strike
23. Fault dip

MODL_blocks.out - summary of block information

```
format(a4, 4f8.3, 2f8.3, 2f7.2, f6.2, 4f6.1, f8.4, f6.1, f7.3, 1x, f8.1,  
14f8.1, i5, 3f6.1, 2f8.2, f8.1)
```

Items:

1. Block name (4-char)
2. Block centroid longitude

3. Block centroid latitude
4. Block pole longitude
5. Block pole latitude
6. Block pole rotation rate (deg/Ma)
7. Block pole rotation rate sigma (deg/Ma)
8. Azimuth of pole error ellipse semi-major axis
9. Pole semi-major axis length (degrees)
10. Pole semi-minor axis length (degrees)
11. Block east velocity at centroid (mm/a)
12. Block north velocity at centroid (mm/a)
13. Block east velocity sigma at centroid (mm/a)
14. Block north velocity sigma at centroid (mm/a)
15. Block velocity NE correlation coefficient at centroid
16. Distance of pole to block centroid (degrees)
17. Vertical axis rotation rate at centroid (deg/Ma)
18. Horizontal velocity gradient at centroid (nanoradians/a)
19. Principle axis of strain rate in block (nanostrain/a; most contractional)
20. Sigma of principle axis of strain rate in block (nanostrain/a)
21. Principle axis of strain rate in block (nanostrain/a; least contractional)
22. Sigma of principle axis of strain rate in block (nanostrain/a)
23. Azimuth of most contractional principle axis of strain rate in block
24. Sigma of azimuth of most contractional principle axis of strain rate in block
25. Principle axis of residual strain rate in block (nanostrain/a; most contractional)
26. Sigma of principle axis of residual strain rate in block (nanostrain/a)
27. Principle axis of residual strain rate in block (nanostrain/a; least contractional)
28. Sigma of principle axis of residual strain rate in block (nanostrain/a)
29. Azimuth of most contractional principle axis of residual strain rate in block
30. Sigma of azimuth of most contractional principle axis of residual strain rate in block
31. Rotation rate of GPS residuals in block (nanoradians/a)
32. Sigma of rotation rate of GPS residuals in block (nanoradians/a)
33. Number of GPS observations in block
34. Not used
35. Not used
36. Angular distance from block centroid to block pole
37. Normalized RMS of GPS in block
38. Weighted RMS of GPS in block
39. Probability that GPS is satisfied (Q of Press et al., 1989; pp. 502-503)

MODL.nod Summary of node information

format (a10, 3i3, 2(1x,a4), 1x, 3f9.3, 2f10.3, 4f8.1, f8.4, 2f8.3, 6f8.1, 1pe10.3)

Item:

1. Fault name (10-char)
2. Fault number
3. Node X index
4. Node Z index
5. Hanging wall block name
6. Foot wall block name

7. Node Longitude
8. Node Latitude
9. Node depth
10. Node phi
11. Node phi uncertainty
12. Fault East slip rate (mm/yr)
13. Fault North slip rate (mm/yr)
14. Fault East slip rate sigma (mm/yr)
15. Fault North slip rate sigma (mm/yr)
16. Fault slip rate NE correlation
17. East component of slip deficit rate (mm/yr)
18. North component of slip deficit rate (mm/yr)
19. Azimuth of slip at node
20. Along strike distance of node (from first node) in km
21. Across strike (horizontal) distance of node from surface node updip from it, in km
22. Downdip distance of node from surface node updip from it, in km
23. Fault strike at node
24. Fault dip at node
25. Moment (rate) associated with this node in N-m

MODL_pNN.out (NN=profile number) contain calculated and observed values along profile lines.

In columns:

First column is a letter:

- C for calculated values,
- G for observed gps,
- T for observed tilt rate,
- U for observed uplift,
- S for observed slip vector azimuth,
- R for observed slip rate,
- V for volcano,
- Q for earthquake,
- L for label

'C' lines (calculated values):

1. C
2. Longitude
3. Latitude
4. Distance along profile (km)
5. Velocity in x
6. Velocity in y
7. Velocity in z
8. Horizontal velocity
9. Radial velocity (along profile line)
10. Transverse velocity (perpendicular to profile)
11. Azimuth of vector
12. Tilt rate in nanoradians/year
13. Radial component of rotation
14. Transverse component of rotation
15. Radial component of locking
16. Transverse component of locking
17. Radial component of strain velocity
18. Transverse component of strain velocity

'G' lines (GPS):

1. G
2. Longitude
3. Latitude
4. Distance along profile (km)
5. E Velocity
6. E sigma
7. N Velocity
8. N sigma
9. Horizontal velocity
10. Horizontal sigma
11. Radial velocity (along profile line)
12. radial sigma
13. Transverse velocity (perpendicular to profile)
14. Transverse sigma
15. Azimuth
16. Azimuth sigma
17. Ve residual
18. Vn residual
19. Normal distance from profile line (km)
20. Site name
21. GPS file name
22. Block name

'U' lines (uplift rates):

1. U
2. Longitude
3. Latitude
4. Distance along profile (km)
5. Up rate observed
6. Up rate sigma
7. Calculated Up rate

'R' lines (slip rates):

1. R
2. Longitude
3. Latitude
4. Distance along profile (km)
5. Slip rate observed
6. Slip rate sigma
7. Calculated slip rate

'T' lines (tilt rates):

1. T
2. Longitude
3. Latitude
4. Distance along profile (km)
5. Tilt rate observed
6. Tilt rate sigma
7. Calculated tilt rate

'S' lines (slip vectors):

1. A
2. Longitude
3. Latitude
4. Distance along profile (km)

5. Azimuth
6. Azimuth sigma

'V' lines (volcanoes):

1. V
2. Longitude
3. Latitude
4. Distance along profile (km)

'Q' lines (earthquakes):

1. Q
2. Longitude
3. Latitude
4. Distance along profile (km)
5. Depth (km)

'L' line (profile label):

The last line contains the label for plotting. Use `grep "^L'` to extract it.

Line 1 242.5 31.8 46 40

It contains in order the Line number, longitude and latitude of starting point, the profile azimuth, and the width of the included data.

MODL_grid_N.vec file contains vectors at grid points in psvelo format. N is the grid number

MODL_grid_N.info file contains grid information

`format(2f9.3, 19(1x,f10.2), 2x, a4, a1)`

1. Longitude of grid point
2. Latitude of grid point
3. East velocity total
4. North velocity total
5. East velocity sigma
6. North velocity sigma
7. E component of block rotation
8. N component of block rotation
9. E component velocity sigma
10. N component velocity sigma
11. NE correlation of block velocity
12. E component of block strain
13. N component of block strain
14. E component of block strain sigma
15. N component of block strain sigma
16. E component of fault locking strain
17. N component of fault locking strain
18. E component of fault locking strain sigma
19. N component of fault locking strain sigma
20. Up component of fault locking strain
21. Up component of fault locking strain sigma
22. Block name (4-char)
23. * if point falls on a fault, blank otherwise

MODL.summary - model statistics by data, block, and pole

First line:

```
format('SUM',1x, a4, 1x, a12, f9.3, 3i6, f10.2, f8.3)
```

1. SUM
2. Model name
3. Run date (YYYYMMDDHHMM)
4. Reduced χ^2
5. Degrees of freedom
6. Number of data
7. Number of free parameters
8. Total χ^2
9. Probability of fit

For each data type/file

```
format(1x,a4, i5, 2(1x, e10.3), 3f9.3, f9.2)
```

1. Data aggregate name (Data type, GPS file code, block code, or pole number)
2. #obs = Number of GPS components
3. DataVar = Weighted variance of data \rightarrow SUM [Obs/Sigma) **2]
4. Chi2 = χ^2 of residual \rightarrow SUM [(Res/Sigma) **2]
5. Chi2/N = χ^2 per observation
6. Nrms = Normalized rms \rightarrow SQRT (χ^2 / N)
7. Wrms = weighted rms \rightarrow SQRT (χ^2 / SumWt)
8. SumWt = sum of weights \rightarrow SUM (1.0/ sigma **2)

MODL.svs - fits to slip vector / transform azimuths

```
format(2f9.3, 4f6.1, f7.2, 2(1x, a4), 1x, a4, 1x, a25,1x,a30)
```

Items:

1. Longitude
2. Latitude
3. Azimuth
4. Sigma
5. Calculated
6. Residual
7. residual / sigma
8. Fixed block
9. Moving block
10. Input file number
11. Slip vector label
12. Input file name

MODL.srs - fits to fault slip rates

```
format(2f9.3, 5f6.1, f7.2, 1x, a1, 1x, 2(1x, a4), 2f6.1, 1x, a4, 1x, a25, 1x, a30)
```

Items:

1. Longitude
2. Latitude
3. Observed slip rate (mean or minimum)
4. Observed slip rate (zero or maximum)
5. Sigma
6. Calculated
7. Residual
8. Residual divided by sigma

DEFNODE User's Manual Version 2007.10.25

9. M* or S** (M indicates min/max was fit, S indicates mean/sigma was fit)
10. Block 1
11. Block 2
12. Azimuth of slip rate measurement
13. Total rate at that point (not corrected for azimuth)
14. Input file number
15. Data label
16. input file name

* If item #9 is M then #3 is the minimum slip rate, #4 is the maximum slip rate, and #5 is the assigned uncertainty for the inversion (one-quarter the min/max range). The residual is zero if the calculated slip rate #6 falls between the min and max observed. If the calculated is outside the observed range, the residual is how far it falls outside the range.

** If item 9 is S, then #3 is the mean slip rate, #4 is not used, and #5 is the assigned uncertainty in the slip rate.

MODL.poles - rotation poles. This file lists all the relative rotation poles as lat, lon, omega and as Cartesian vectors with uncertainties.

```
format (2i3, 1x, a4, 3f10.4, 6f10.4)
```

1. File number
2. No = pole number
3. Name = Block/GPS file name
4. Wx = x component of angular velocity (deg/Myr)
5. Wy = y component of angular velocity (deg/Myr)
6. Wz = z component of angular velocity (deg/Myr)
7. Sx = standard error of Wx
8. Sy = standard error of Wy
9. Sz = standard error of Wz
10. Sxy = covariance of Wx and Wy
11. Sxz = covariance of Wx and Wz
12. Syz = covariance of Wy and Wz

and

```
format (2i3, 1x, a4, 4f10.4, ff8.2)
```

1. File number
2. No = pole number
3. Name = Block/GPS file name
4. Lon. = Longitude of pole
5. Lat. = Latitude of pole
6. Omega = rotation rate (deg/Myr)
7. SigOm = standard error of rotation rate (deg/Myr)
8. Emax = maximum axis of lon/lat error ellipse
9. Emin = minimum axis of lon/lat error ellipse
10. Az = azimuth of maximum axis of lon/lat error ellipse

MODL.strain - strain rate tensors in nanostrain/year

Block strain rate tensors in N-E coordinate system

```
format (a1,i4, 1x, a4, 2f8.3, 6f10.2, 3f10.4)
```

1. T

DEFNODE User's Manual Version 2007.10.25

2. No = pole number
3. Block = Block name
4. Longitude of origin
5. Latitude of origin
6. Exx = normal strain rate in East direction
7. Eyy = normal strain rate in North direction
8. Exy = shear strain rate
9. Sxx = standard error in Exx
10. Syy = standard error in Eyy
11. Sxy = standard error in Exy
12. Cxx-yy = covariance between Exx and Eyy
13. Cxx-xy = covariance between Exx and Exy
14. Cyy-xy = covariance between Exy and Eyy

Block principle strain rates (nanostrain/yr)
format(a1,i4, 1x, a4, 2f8.3, 6f10.2)

1. P
2. No = pole number
3. Block = Block name
4. Longitude of origin
5. Latitude of origin
6. E1 = most compressive strain rate
7. SigE1 = standard error in E1
8. E2 = least compressive strain rate
9. SigE2 = standard error in E2
10. A1 = azimuth of E1
11. SigA1 = standard error in A1

Block residual principle strain rates (ns/yr) and rotations (nanoradians/yr);
format(a4, 8f10.2)

1. Block = Block name
2. E1 = most compressive strain rate
3. SigE1 = standard error in E1
4. E2 = least compressive strain rate
5. SigE2 = standard error in E2
6. A1 = azimuth of E1
7. SigA1 = standard error in A1
8. Rot = rotation rate in nanoradians/year
9. SigRot = standard error in Rot

MODL.hc - results of hard constraints

format(a4, i2, 2f10.4, 2(1x,a4), 3f8.2, 1x, e12.4)

Item

1. Type
2. Type code: 1 = slip rate; 2 = azimuth; 3 = rotation
3. Longitude
4. Latitude
5. Fixed block
6. Moving block
7. Minimum value
8. Maximum value
9. Calculated value
10. Penalty

MODL.table - misc results

```
format(a1, a8, 1x, a14, 2f9.3, 4f7.1, f8.4, 14f7.1, f8.2)
```

Item

1. Used = 'x' if datum not used
 2. Site = site name
 3. Srce = GPS velocity field
 4. Blck = block name
 5. Pole = pole number
 6. Longit. = longitude
 7. Latit. = latitude
 8. Ve = observed East velocity (corrected for ref frame rotation)
 9. Vn = observed North velocity (corrected for ref frame rotation)
 10. Se = standard error in East velocity
 11. Sn = standard error in North velocity
 12. Sne = NE correlation
 13. Enet = East velocity from ref frame rotation
 14. Nnet = North velocity from ref frame rotation
 15. Eref = Obs East velocity in reference frame
 16. Nref = Obs North velocity in reference frame
 17. Eela = East velocity component of elastic deformation
 18. Nela = North velocity component of elastic deformation
 19. Erot = East velocity component of block rotation
 20. Nrot = North velocity component of block rotation
 21. Estr = East velocity component of permanent strain
 22. Nstr = North velocity component of permanent strain
 23. Eres = East velocity residual
 24. Nres = North velocity residual
 25. Vres = Total residual rate
 26. Vsig = Total residual sigma
 27. R/S = Vres / Vsig
-

PLOTTING WITH GMT

Using GREP and AWK to make profiles

Use *grep* and *awk* to extract desired columns from the profile files. For example, to get the profile distance and the observed East GPS velocity and sigma:

```
% grep '^G' MODL_p01.out | awk '{ print $4, $5, $6 }' | psxy ...
```

grep '^G' gets all the lines starting with 'G' from the file, the 'awk' command prints the 4th, 5th, and 6th entries from each line.

How to plot maps. Most things are accessible for plotting with GMT. Some may require clever scripting. Here are a few suggestions.

Vector files are generally in *psvelo -Se* format

```
psvelo MODL.vec -Se0.03/0.7/7 ...
```

but have other info appended to the line. If *psvelo* cannot handle the long line, use *cut*

```
cat MODL.res | cut -c1-65 | psvelo -Se ...
```

OR *awk*

```
awk '{ print $1, $2, $3, $4, $5, $6, $7, $8 }' MODL.res | psvelo -Se ...
```

The MODL_flt_atr.gmt file contains fault attributes and quadrilaterals and can be used to make color plots. Since the header line for each segment (fault) has multiple attributes following the -Z, use *awk* to select the one to plot.

```
awk '{ if ($1 == ">") print $1,$2,$4; else print $1,$2 }' MODL_flt_atr.gmt | psxy -Cpalette.cpt ....
```

where \$1 = '>', \$2 = '-Z'

and the attributes are (in order): (3)fault_number, (4)slip_rate_deficit, (5)phi, (6)phi_error, (7)slip_rate, (8)fault_parallel_slip_rate, (9)fault_normal_slip_rate

The MODL_blk.gmt file, containing the block outlines, similarly has multiple attributes on the header line; the attributes are (in order): (3)block_number, (4)pole_number, (5)block_name

filled blocks, fill color based on pole number

```
awk '{ if ($1 == ">") print $1,$2,$4; else print $1,$2 }' MODL_blk.gmt | psxy -Cpalette.cpt -L -M ...
```

unfilled block outlines

```
psxy MODL_blk.gmt -W5/100/100/100 -L -M ...
```

dashed lines for all profiles

```
psxy MODL_lin.gmt -R -J -W4/0/200/200t5_5:5 -M ...
```

or, for a single profile (#19 in this case),

```
grep '^C' MODL_p19.out | awk '{ print $2, $3 }' | psxy -W4/0/200/200t5_5:5 ...
```

dots at all node positions

```
awk '{print $7, $8}' MODL.nod | psxy -Sc.1i ....
```

dots at surface node positions

```
awk '{if ($4==1) print $7, $8}' MODL.nod | psxy -Sc.1i ...
```

#label node with fault number at surface only

```
awk '{if ($4==1) print $7, $8, " 7 0 0 CM ", $2}' MODL.nod | pstext ....
```

label blocks with names (-W255/255/255 results in whiting out beneath label)

```
awk '{ print $2, $3, " 8 0 0 CM ", $1 }' MODL_blocks.out | pstext -W255/255/255 ...
```

```
# plot pole positions (dot) and error ellipses
awk ' { print $4, $5 } ' MODL_blocks.out | psxy -Sc0.1i ....
awk ' { print $4, $5, $8, $9*111.2, $10*111.2 } ' MODL_blocks.out | psxy -SE ....

# plot fault slip vectors halfway between fault nodes
awk '{ print $3, $4, $5, $6, $7, $8, $9, $10 }' MODL_mid.vec | psvelo -Se ....

# principle axes for block strain rates
awk ' { print $2, $3, $21, $19, $23 } ' MODL_blocks.out | psvelo -Sx0.1 ...
```

SAMPLE INPUT

See the file cr.dfn for an example of the control file.

Acknowledgments: Thanks to C. Williams, Y. Okada, S. Roecker, and C. DeMets for supplying various subroutines. And to L. Wallace and L. Prawirodirdjo for testing the program. Program development was supported by NSF and USGS NEHRP grants.

CITATIONS

- DeMets, C., R. G. Gordon, D. F. Argus, and S. Stein (1990). Current plate motions, *Geophys. J. Int.* **101**, 425-478.
- McCaffrey, R., M. D. Long, C. Goldfinger, P. Zwick, J. Nabelek, and C. Smith (2000). Rotation and plate locking at the southern Cascadia subduction zone, *Geophysical Research Letters*, **27**, 3117-3120. ([PDF](#))
- McCaffrey, R. (2002). Crustal block rotations and plate coupling, in Plate Boundary Zones, Geodynamics Series 30, S. Stein and J. Freymueller, editors, 101-122, AGU. ([PDF](#))
- McCaffrey, R., (2005). Block kinematics of the Pacific - North America plate boundary in the southwestern US from inversion of GPS, seismological, and geologic data, *Journal of Geophysical Research* **110**, B07401, doi:10.1029/2004JB003307, 2005.
- McCaffrey, R., A. I. Qamar, R. W. King, R. Wells, Z. Ning, C. A. Williams, C. W. Stevens, J. J. Vollick, and P. C. Zwick, (2007). Plate locking, block rotation and crustal deformation in the Pacific Northwest, *Geophysical Journal International*, doi:10.1111/j.1365-246X.2007.03371.x.
- Okada, Y. (1985). Surface deformation to shear and tensile faults in a half-space, *Bull. Seismol. Soc. Am.*, **75**, 1135-1154.
- Okada, Y. (1992). Internal deformation due to shear and tensile faults in a half-space, *Bull. Seismol. Soc. Am.*, **82**, 1018-1040.
- Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling (1989). *Numerical Recipes*, Cambridge Univ. Press, Cambridge.

- Savage, J. C., W. Gan, and J. L. Svarc (2001). Strain accumulation and rotation in the Eastern California Shear Zone, *Journal of Geophysical Research*, **106**, 21995-22071.
- Wang, K., Wells, R., Mazzotti, S., Hyndman, R. D., & Sagiya, T. (2003). A revised dislocation model of interseismic deformation of the Cascadia subduction zone, *J. Geophys. Res.*, **108**, 2026, doi:10.1029/2001JB001227.
- Wessel, P., and W. H. F. Smith (1991). Free software helps map and display data, *EOS Trans. AGU*, **72**, 445-446.

Papers using DEFNODE:

- Prawirodirdjo, L., Y. Bock, R. McCaffrey, J. Genrich, E. Calais, C. Stevens, S. S. O. Puntodewo, C. Subarya, and J. Rais (1997), Geodetic observations of interseismic strain segmentation at the Sumatra subduction zone, *Geophysical Research Letters*, 2601-2604. ([html](#)) ([PDF](#))
- McCaffrey, R., M. D. Long, C. Goldfinger, P. Zwick, J. Nabelek, and C. Smith, Rotation and plate locking at the southern Cascadia subduction zone (2000), *Geophysical Research Letters*, **27**, 3117-3120. ([PDF](#))
- McCaffrey, R. (2002), Crustal block rotations and plate coupling, in Plate Boundary Zones, Geodynamics Series 30, S. Stein and J. Freymueller, editors, 101-122, AGU. ([PDF](#))
- Wallace, L. M., C. W. Stevens, E. Silver, R. McCaffrey, W. Loratung, S. Hasiata, R. Curley, R. Rosa, J. Taugaloidi, and H. Davies (2004), GPS Constraints on Active Tectonics and Arc-Continent Collision in Papua New Guinea: evidence for edge-driven microplate rotations, *Journal of Geophysical Research* **109**, B05404, doi:10.1029/2003JB002481. ([PDF](#))
- Wallace, L. M., J. Beavan, R. McCaffrey, and D. Darby (2004), Subduction Zone Coupling and Tectonic Block Rotations in the North Island, New Zealand, *Journal of Geophysical Research*, **109**, B12406, doi:10.1029/2004JB003241. ([PDF](#))
- McCaffrey, R. (2005), Block kinematics of the Pacific - North America plate boundary in the southwestern US from inversion of GPS, seismological, and geologic data, *Journal of Geophysical Research* **110**, B07401, doi:10.1029/2004JB003307. ([PDF](#) and [Supplemental data](#)).
- Wallace, L.M., R. McCaffrey, J. Beavan, and S. Ellis (2005). Rapid microplate rotations and backarc rifting at the transition between collision and subduction, *Geology* **33**, 857-860. ([PDF](#))
- Vigny, C. et al. (2005). Insight into the 2004 Sumatra-Andaman earthquake from GPS measurements in southeast Asia *Nature* **436**, 201-206.
- Subarya, C., M. Chlieh, L. Prawirodirdjo, J.-P. Avouac, Y. Bock, K. Sieh, A. J. Meltzner, D. H. Natawidjaja, and R. McCaffrey (2006). Plate-boundary deformation of the great Aceh-Andaman earthquake, *Nature* **440**, p. 46-51, doi10.1038. ([PDF](#))
- Socquet, A., W. Simons, C. Vigny, R. McCaffrey, C. Subarya, D. Sarsito, B. Ambrosius, W. Spakman (2006), Micro block rotations and fault coupling in SE Asia triple junction (Sulawesi, Indonesia) from GPS and Earthquake Slip Vector data, *Journal of Geophysical Research*, **111**, B08409, doi:10.1029/2005JB003963.
- Wallace, L.M., J. Beavan, R. McCaffrey, K. Berryman, and P. Denys (2006). Balancing the plate motion budget in the South Island, New Zealand using GPS, geological and

seismological data, *Geophysical Journal International*, , doi:10.1111/j.1365-246X.2006.03183.x. ([PDF](#))

- Socquet, A., C. Vigny, N. Chamot-Rooke, W. Simons, C. Rangin, and B. Ambrosius (2006), India and Sunda plates motion and deformation along their boundary in Myanmar determined by GPS, *J. Geophys. Res.*, 111, B05406, doi:10.1029/2005JB003877.
- Galgana, G., M. Hamburger, R. McCaffrey, E. Corpuz, and Q. Chen (2007). Crustal deformation in Luzon Island, Philippines using terrain models, geodetic observations and focal mechanisms, *Tectonophysics*, 432, 63-87. ([PDF](#))
- McCaffrey, R., A. I. Qamar, R. W. King, R. Wells, G. Khazaradze, C. A. Williams, C. W. Stevens, J. J. Vollick, and P. C. Zwick (2007). Fault locking, Block Rotation and Crustal Deformation in the Pacific Northwest, *Geophysical Journal International*, doi:10.1111/j.1365-246X.2007.03371.x. ([PDF](#); [Velocity field](#); [Supplement](#)).
- Plattner, C., R. Malservisi, T. H. Dixon, P. LaFemina, G. F. Sella, J. Fletcher and F. Suarez-Vidal (2007). New constraints on relative motion between the Pacific Plate and Baja California microplate (Mexico) from GPS measurements, *Geophys. J. Int.*, doi: 10.1111/j.1365-246X.2007.03494.

Theses:

- Prawirodirdjo, L. M., (2000 PhD UCSD) A Geodetic Study of Sumatra and the Indonesian Region: Kinematics and Crustal Deformation from GPS and Triangulation.
- Wallace, L.M. (2002 PhD UCSC), Tectonics and arc-continent collision in Papua New Guinea: Insights from geodetic, geophysical and geologic data.
- Galgana, G. (2005 Master's Indiana University) Kinematics of an active plate boundary zone: insights on the tectonics of Luzon, Philippines using terrain models, focal mechanisms and GPS observations.