Computer Code for USGS Open-File Report 2012–1226

Thermodynamic Method for Generating Random Stress Distributions on an Earthquake Fault

by Michael Barall (Invisible Software, Inc.)
and Ruth A. Harris (U.S. Geological Survey)

This computer code implements the method for generating random heterogeneous stress on an earthquake fault, using principles from thermodynamics.

Version 1.00

Contents

1. Compiling and Running the Code

The code can be used on Unix/Linux, Macintosh, and Windows computer systems.

To compile the code, you need a Fortran 95 (or later version of Fortran) compiler. You should compile and link the following two files:

genths.f
mt19937ar.f

The first file contains the main program, and the second file contains the random number generator.

When compiling, you should set compiler options so that the Fortran INTEGER data type is 64-bits or 8-bytes in size. Refer to your compiler documentation for the correct option to use.

The program is intended to be executed from a command prompt. When you run the program, it reads its parameters from the standard input. You should write a text file containing the desired parameters, and then redirect standard input to read from the text file. For example, if the compiled program is named genths (or genths.exe on Windows), you could execute it like this:

```
genths  <input_file
```

In this example, input_file is the name of a text file containing input parameters, and the < symbol means to read standard input from that file.

2. Overview and Running the Sample Input

The purpose of this code is to generate random distributions of stress on a fault surface, suitable for use as input to a dynamic rupture simulation. The end result of running the code is an output file, which specifies how stress varies as a function of position on the fault. We lay out a rectangular array of nodes on the fault surface (see section 3). For each node, the output file contains shear and normal stresses on the fault at the node's location. The file also contains friction parameters for the node. The output file format is described in section 7.

Each pattern of random stresses is called a *realization*. We adopt the convention of identifying a realization by two integers, a *realization number* and a *subrealization number*. We write the two numbers connected by an underscore, so for example, 00200_007 refers to realization 200 and subrealization 7. The realization number identifies the seed value that is used to initialize the random number generator. When running the program, you must supply a text file that contains a list of seed values. Realization 200 is generated using the seed value on line 200 of the seed file. Refer to section 9 for further information on random number generation.

The computation is done in two phases, and so it is necessary to run the program at least twice. In phase 1, the program performs the following tasks:

- Use the thermodynamic algorithm to generate random, raw stress values over a very large area.
- Use the selection algorithm to select a sub-area the size of the fault surface.
- Smooth the raw stress distribution with a moving-average filter.

In phase 2 the program performs these tasks:

- Scale a group of realizations, to produce scaled stress values that are consistent with the assumed friction law.
- Calculate friction parameters.
- Add information to artificially nucleate the rupture.

The program must be run one or more times to perform phase 1, and then run again to perform phase 2. The parameters for phase 1 are described in section 5, and the parameters for phase 2 are described in section 6.

The sample input consists of two input files for phase 1 (tpvx05_gen_200.txt and tpvx05_gen_201.txt), one input file for phase 2 (tpvx05_fric_200-201.txt), and a small random seed file (randseed.txt). Using the sample input, you first generate raw stress for realizations 00200_000 through 00200_009 with the command:

```
genths  <tpvx05_gen_200.txt
```

Then, you generate raw stress for realizations 00201_000 through 00201_009 with:

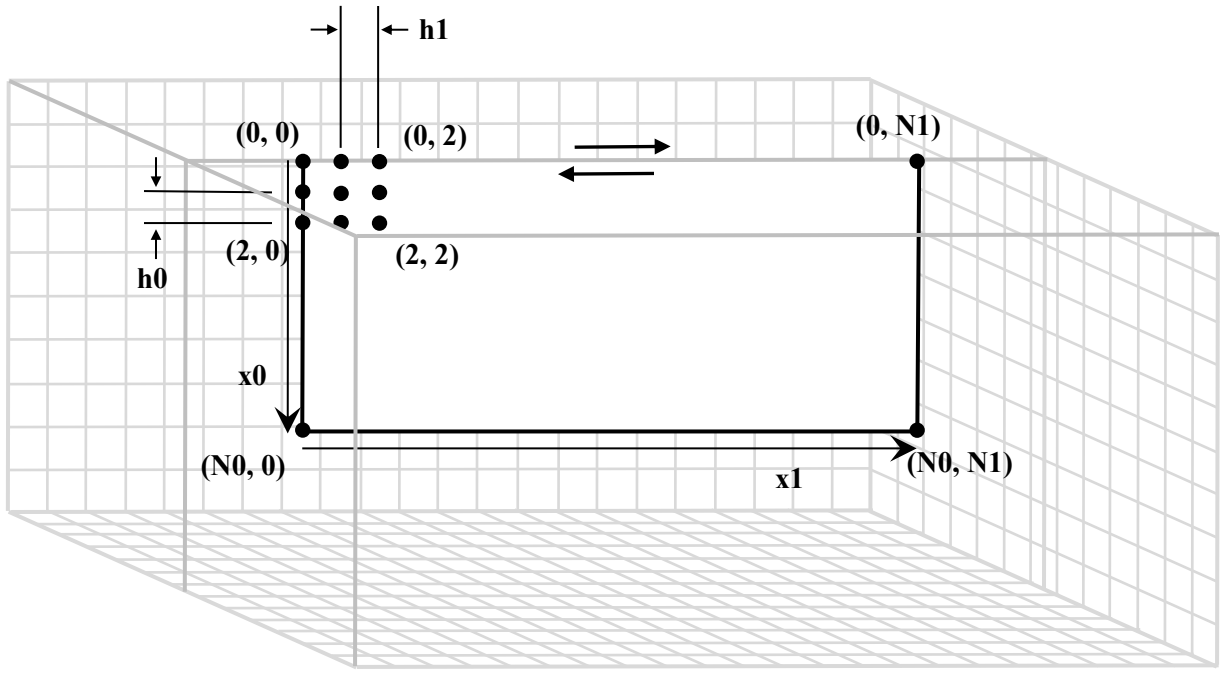
```
genths  <tpvx05_gen_201.txt
```

Both of the above commands look in randseed.txt to obtain the seed for the random number generator. Finally, you compute the scaled stresses and friction parameters for all twenty realizations 00200_000 through 00200_009 and 00201_000 through 00201_009 with:

```
genths  <tpvx05_fric_200-201.txt
```

Note that in order to use the sample input files, you need about 42 gigabytes of available computer memory. If you don't have that much memory, then you need to edit the input files for phase 1 and reduce the size of the large Fourier domain.

3. Fault Layout

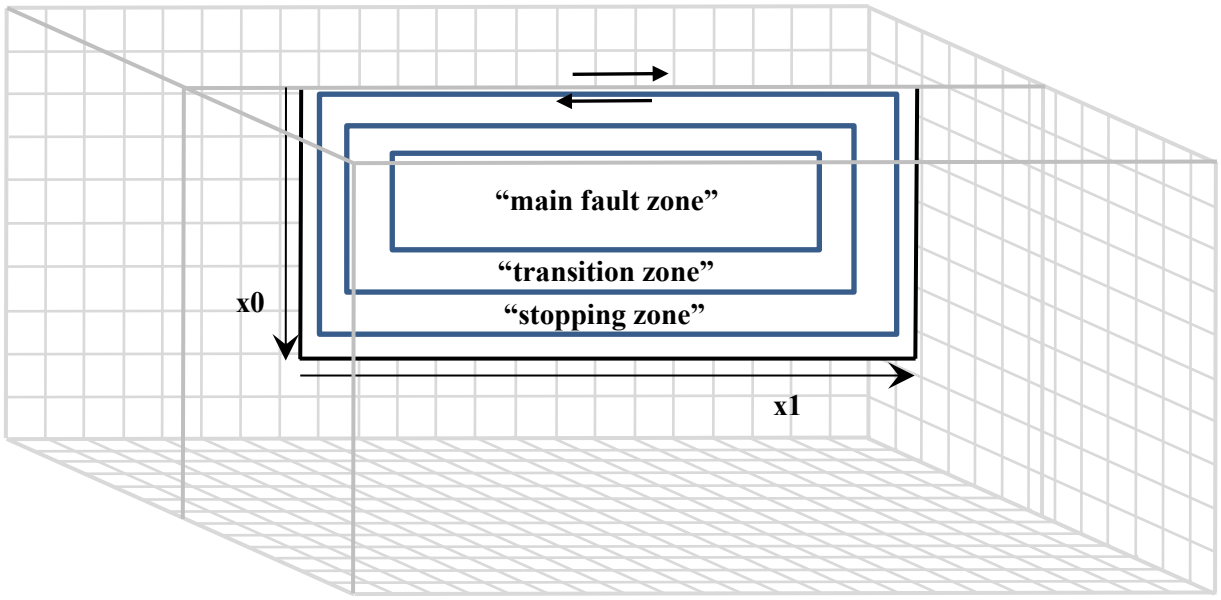


The above figure shows the layout of nodes and cells on the fault surface. The program calculates stresses and friction parameters for each node on the fault surface.

Positions on the fault surface are denoted by coordinates (x_0, x_1). The x_0 coordinate is vertical or along-dip, and increases from top to bottom, with $x_0 = 0$ at the earth's surface. The x_1 coordinate runs along-strike, and increases from right to left, with $x_1 = 0$ at the left edge of the fault. Notice that for convenience, in this document the coordinates x_0 and x_1 have opposite sign from the coordinates in the open-file report.

A rectangular array of *nodes* is laid out on the fault surface, which partitions the fault surface into a set of rectangular *cells*. Nodes are equally spaced, by distances h_0 and h_1 in the along-dip and along-strike directions respectively. Each node is denoted by integer coordinates (n_0, n_1) where $0 \le n_0 \le N_0$ and $0 \le n_1 \le N_1$. So, the total number of nodes is $(N_0 + 1)(N_1 + 1)$.

Each cell is identified by the node in its upper left corner. So, a cell can be identified by coordinates (n_0, n_1) where $0 \le n_0 < N_0$ and $0 \le n_1 < N_1$. The total number of cells is $N_0 N_1$.



The above figure shows three *zones* on the fault surface, which are used in scaling the randomly-generating stresses. The zones are defined by three nested rectangles, shown in blue.

The interior of the innermost rectangle is called the "main fault zone." This zone is where we attempt to make the initial stress high enough so that the rupture can propagate.

The region between the outermost rectangle and the middle rectangle is called the "stopping zone." This zone is where we attempt to make the initial stress low enough so that the rupture stops propagating.

The region between the middle rectangle and the innermost rectangle is called the "transition zone." This zone is where the stress transitions from high values within the main fault, to low values within the stopping zone.

Remark: Generally, the main fault zone and the stopping zone correspond to the high-value and low-value parts of the selection template, which is used to select the fault surface from a much larger randomly-generated stress distribution (refer to the open-file report for details).

Remark: For a rupture that extends the full depth of the seismogenic zone, generally the outermost rectangle fills the entire fault surface, except perhaps for a region near the bottom of the fault where rupture extent is controlled by depth conditioning. Also, the middle and innermost rectangles generally have the same height as the outermost rectangle.

4. Linear Slip-Weakening Friction Law

We use a linear slip-weakening friction law. Each node on the fault surface is assigned the following five friction parameters:

Friction Parameters		
Symbol	*Parameter*	*Unit*
μ_s	Static coefficient of friction.	Dimensionless
μ_d	Dynamic coefficient of friction.	Dimensionless
D_c	Slip-weakening critical distance.	Meter
C	Cohesion.	Pascal
T	Time of forced rupture.	Second

In addition, each node on the fault surface is assigned initial shear and normal stresses:

Initial Stresses		
Symbol	*Parameter*	*Unit*
τ_0	Initial shear stress, at time $t = 0$.	Pascal
σ_0	Initial normal stress, at time $t = 0$.	Pascal

When the fault is sliding, the shear stress τ at a given point on the fault is related to the normal stress σ by

$$\tau = C + \mu \times \max(0, \sigma)$$

The term $\max(0, \sigma)$ implies that the fault is not permitted to open, and the friction is not allowed to drop below the cohesion, even if normal stress σ becomes negative.

The time-varying coefficient of friction μ is given by:

$$\mu = \begin{cases} \mu_s + (\mu_d - \mu_s) \times D/D_c, & \text{if } D < D_c \text{ and } t < T \\ \mu_d, & \text{if } D \geq D_c \text{ or } t \geq T \end{cases}$$

where D is the total distance the node has slipped and t is the time since the start of the earthquake.

The distance D that the node has slipped is path-integrated. For example, if the node slips 0.4 m in one direction and then 0.1 m in the opposite direction, the value of D is 0.5 m (and not 0.3 m).

The time of forced rupture T is used to nucleate the rupture. At time $t = T$, the node is forced to rupture by setting the coefficient of friction μ equal to the dynamic coefficient of friction μ_d. If a given node does not undergo forced rupture, then the value of T is 1.0E9.

5. Input File Format for Phase 1, Raw Stress Generation

The input file supplies the parameters that the code needs to generate the raw stress field, select the fault surface out of a large random stress distribution, and perform smoothing. When you start the program, you must redirect standard input so the program reads parameters from the input file.

The input file is a text file, which must conform to Fortran input requirements. Each parameter can be an integer, real number, or character string. The program expects each line in the file to contain a certain number of parameters, as described below. Any text that appears on the line after the last expected parameter is treated as a comment, and is ignored. In the sample input files we adopted the convention that comments begin with a # character, however the program does not actually look for the # character. The input file may include blank lines, however, there is no provision for the input file to include a line that contains just a comment.

In this document, we show an example of an input file, accompanied by a description and explanation. The contents of the sample file are shown as **`blue courier text`**. The description and explanation is shown as ordinary black text. The example shown here is the sample input file tpvx05_gen_200.txt.

The first line of the file must contain the character string "run_gen01". This commands the program to generate a raw stress distribution.

```
run_gen01
```

The next line contains the realization number. This number selects the seed which is used to initialize the random number generator. The value 200 selects the seed on line number 200 of the random seed file. Since lines are numbered beginning with zero, this is actually the 201st line of the random seed file.

```
200
```

The next line contains the name of the random seed file. In this case, it is randseed.txt. Each line in the random seed file must contain eleven integers, which are used as a seed to initialize the random number generator. Refer to section 9 for further information.

```
randseed.txt
```

The next line contains the maximum subrealization number to generate. In this example, the maximum subrealization number is 9, which means that the program will generate ten realizations numbered 00200_000 through 00200_009.

```
9
```

The following two character strings are used to construct the names of output files, which contain the raw stress distribution for each generated realization. These files are later read in

during phase 2 of the program. The file format is documented in section 8. Each file name is constructed by inserting the realization and subrealization numbers in between the two character strings, which are used as a prefix and suffix respectively. In this example, there are ten output files named tpvx05_00200_000_raw.txt through tpvx05_00200_009_raw.txt.

```
tpvx05   raw.txt
```

The following two character strings are used to construct the names of output files, which contain some statistical information about each generated realization. Each file name is constructed by inserting the realization and subrealization numbers in between the two character strings, which are used as a prefix and suffix respectively. In this example, there are ten output files named tpvx05_00200_000_gout.txt through tpvx05_00200_009_gout.txt.

```
tpvx05   gout.txt
```

The following two character strings are used to construct the names of an output file, which contains a one-line summary of each generated realization. The file name is constructed by inserting the realization number in between the two character strings, which are used as a prefix and suffix respectively. In this example, the output file is named tpvx05_00200_gsum.txt.

```
tpvx05   gsum.txt
```

The following two integers give the number of cells on the fault surface, in the along-dip and along-strike directions respectively. In this example, the fault has 260 cells along-dip and 640 cells along-strike, or equivalently 261 nodes along-dip and 641 nodes along-strike. See section 3 for a diagram of the fault layout.

```
260   640
```

The following two real numbers give the cell size, or equivalently, the spacing between adjacent nodes, in meters. The first number is the spacing along-dip, and the second number is the spacing along-strike. The example has a spacing of 75 meters in each direction, which gives a total fault size of 75×260 = 19500 meters along-dip, and 75×640 = 48000 meters along-strike.

```
75.0   75.0
```

The following two integers are called the cell stride. They should each be set to 1.

```
1   1
```

The stress distribution is smoothed using a moving-average filter. The following two integers give the radius of the filter mask, measured in cells. The first integer is the radius in the along-dip direction, and the second integer is the radius in the along-strike direction. We recommend setting both values to 3, which results in a filter mask measuring 5×5 nodes. If both values are set to 1, then there is no smoothing.

```
3   3
```

These two real numbers are extra scale factors, which are applied during smoothing. They should be set to 1.0 and 0.0, as shown.

```
1.0  0.0
```

The following integer is set to 1 to generate strike-slip raw stresses, or 0 to generate dip-slip raw stresses. In the current code, it should always be set to 1 as shown. You can adjust the rake angle as desired during phase 2 of the computation.

```
1
```

The next line contains two integers, which are the size of the large Fourier domain. The first integer is the number of nodes along-dip, and the second integer is the number of nodes along-strike. Note that the Fourier domain is much larger than the fault surface. Both integers are rounded up to the next power of two, so the values shown actually request a Fourier domain measuring 32768×32768 nodes. The memory required by the program is approximately proportional to the total number of nodes in the Fourier domain. A 32768×32768 domain requires about 42 gigabytes of available memory, while a 16384×16384 domain requires about 11 gigabytes of available memory.

```
32000  32000
```

The following integer controls the stress orientation (or rake angle) used when forming a linear combination of Fourier components. It should always be 2.

```
2
```

The following integer selects the type of phase shift that is allowed during the selection algorithm, which selects the fault surface from the much larger Fourier domain. A value of 1 allows no phase shift, a value of 2 allows a sign change, and a value of 3 allows an arbitrary complex phase shift. The recommended value is 3.

```
3
```

The following lines give the selection template $Q(x_0, x_1)$ as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. Refer to the open-file report for an explanation of how the selection template is used. You should choose the selection template function to be positive in portions of the fault where you want high stress (so the rupture can propagate), and negative in portions of the fault where you want low stress (so the rupture stops). Zero values can be used in portions of the fault where the stress value is unimportant (for instance, near the bottom of the fault if depth-conditioning is being used). In this example, the selection template is defined to be the following function of two variables:

$$Q(x_0, x_1) \equiv Q_0(x_0) Q_1(x_1)$$

where Q_0 and Q_1 are piecewise linear functions:

$$Q_0(x_0) \equiv \begin{cases} 0, & \text{if } x_0 < -500 \\ L(x_0, -500, 0, 0, 1), & \text{if } -500 \le x_0 < 0 \\ 1, & \text{if } 0 \le x_0 < 16950 \\ L(x_0, 16950, 1, 17450, 0), & \text{if } 16950 \le x_0 < 17450 \\ 0, & \text{if } 17450 \le x_0 \end{cases}$$

$$Q_1(x_1) \equiv \begin{cases} 0, & \text{if } x_1 < -500 \\ L(x_1, -500, 0, 0, -1), & \text{if } -500 \le x_1 < 0 \\ -1, & \text{if } 0 \le x_1 < 6000 \\ L(x_1, 6000, -1, 12000, 1), & \text{if } 6000 \le x_1 < 12000 \\ 1, & \text{if } 12000 \le x_1 < 36000 \\ L(x_1, 36000, 1, 42000, -1), & \text{if } 36000 \le x_1 < 42000 \\ -1, & \text{if } 42000 \le x_1 < 48000 \\ L(x_1, 48000 - 1, 48500, 0), & \text{if } 48000 \le x_1 < 48500 \\ 0, & \text{if } 48500 \le x_1 \end{cases}$$

and where the function L denotes linear interpolation between two points:

$$L(x, X_0, Y_0, X_1, Y_1) \equiv ((x - X_0)Y_1 + (X_1 - x)Y_0)/(X_1 - X_0)$$

Notice that in this example, nonzero values of Q extend up to 500 meters outside the fault surface (which extends from 0 to 48000 meters along-strike, and 0 to 19500 meters along-dip), so that Q can be tapered to zero rather than changing to zero discontinuously. Also, Q is chosen to be zero in the bottom 2050 meters of the fault surface, because depth conditioning forces the rupture to stop near the bottom of the fault surface, so there is no reason to have a negative Q there.

(The lines below can be read as follows. The initial integer 2 indicates that the function Q is a product of two functions of one variable. The next two lines contain six real numbers that specify a linear transformation, in this case the identity transformation. The next line contains an offset value that is added to the product, in this case zero. The next integers 2 and 4 indicate that Q_0 is a piecewise-linear function with four control points. The next four lines are the four control points for Q_0, each containing a value of x_0 followed by the values of Q_0 immediately before and after the control point. The next integers 2 and 8 indicate that Q_1 is a piecewise-linear function with eight control points. The next eight lines are the eight control points for Q_1, each containing a value of x_1 followed by the values of Q_1 immediately before and after the control point. In this example, the function values immediately before and after each control point are equal, which makes the function continuous; unequal values would specify a discontinuous function.)

```
2
1.0   0.0   0.0
0.0   1.0   0.0
0.0
```

```
2
4
   -0.50e3    0.0    0.0
    0.00e3    1.0    1.0
   16.95e3    1.0    1.0
   17.45e3    0.0    0.0
2
8
   -0.5e3    0.0    0.0
    0.0e3   -1.0   -1.0
    6.0e3   -1.0   -1.0
   12.0e3    1.0    1.0
   36.0e3    1.0    1.0
   42.0e3   -1.0   -1.0
   48.0e3   -1.0   -1.0
   48.5e3    0.0    0.0
```

The next line contains a flag which indicates whether the selection template function $Q(x_0, x_1)$ should be modified to force its DC component to be exactly zero. A value of 1 forces a zero DC component, while a value of 0 instructs the code to use Q without modification. We recommend using a value of 1 as shown.

```
1
```

The next line contains a flag which indicates whether to enable fault surface rotation. A value of 1 enables rotation, while a value of 0 disables rotation. If you enable rotation, then you must choose a selection template function $Q(x_0, x_1)$ which is symmetric under rotation by 180 degrees. In this case, when the code selects the fault surface from the large Fourier domain, it can select either an un-rotated fault surface or a fault surface that is rotated by 180 degrees. The choice is made using the rotation template function $R(x_0, x_1)$, in the manner described in the open-file report.

```
1
```

The following line is present only if fault surface rotation is enabled. (If rotation is disabled, the line should be omitted from the input file.) It contains two integers, which gives twice the displacement from the center of the fault surface to the center of symmetry of the selection template Q, measured in cells. The first integer is twice the displacement in the along-dip direction, and the second integer is twice the displacement in the along-strike direction. In the example shown, the center of symmetry of the selection template $Q$is located 17 cells (that is, 1275 meters) above the center of the fault surface.

```
-34   0
```

The following lines are present only if fault surface rotation is enabled. (If rotation is disabled, the lines should be omitted from the input file.) The following lines give the rotation template $R(x_0, x_1)$ as a function of position (x_0, x_1) on the fault surface. The lines represent a function of

two variables, in the manner described in section 11. Refer to the open-file report for an explanation of how the rotation template is used. In choosing between the un-rotated and rotated fault surface, the code prefers the choice which has higher stress in areas where R is large. In the example shown, $R(x_0, x_1)$ equals 1 in the rectangular area $11000 \le x_0 \le 14500$ and $14000 \le x_1 \le 34000$, and tapers off to 0 elsewhere. The reason for choosing R to be large in the lower half of the fault surface is that nucleation tends to occur there.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
4
    4.0e3   0.0   0.0
   11.0e3   1.0   1.0
   14.5e3   1.0   1.0
   15.5e3   0.0   0.0
2
4
   12.0e3   0.0   0.0
   14.0e3   1.0   1.0
   34.0e3   1.0   1.0
   36.0e3   0.0   0.0
```

The next four lines specify the energy rolloff function. As explained in the open-file report, the energy rolloff function is $g(k) = 1/(1 + (\eta k)^3(\log(1 + \zeta k))^2)$ where η and ζ are constants with dimension of meters. The purpose of these four lines is to select values for η and ζ, but this is done somewhat indirectly.

The first line must contain the integer 2, which tells the program to use the log-square rolloff formula. The second line contains the correlation length Λ in meters. The third line contains a value we call b which has dimension of meters, and the fourth line contains a value we call m which is dimensionless. The code then calculates η and ζ to satisfy the following two conditions: (1) the function $k^2 g(k)$ reaches its maximum value at $k = 2\pi/\Lambda$; and (2) $\zeta = m\eta + b/2\pi$.

We have found that the correlation length Λ should be about the desired size of the rupture. In the example shown, the correlation length is 30000 meters. This is equal in size to the main fault zone plus half the transition zone. (As shown later, the main fault zone is 24000 meters along-strike, and the transition zone adds an additional 12000 meters along-strike. See the figure in section 3.) In the example shown, $b = 0$ and $m = 1$, which implies that $\zeta = \eta$.

```
2
30000.0
    0.0
1.0
```

The next six lines specify the three rectangular zones on the fault surface: the main fault zone, the transition zone, and the stopping zone. See the figure in section 3. Generally, the main fault zone and the stopping zone correspond to the high-value and low-value parts of the selection template. The code uses these zones to calculate statistical properties of the different portions of the fault surface.

Each line contains two integers, which represent a range of cells. Recall from section 3 that there is a rectangular array of nodes on the fault surface, identified by integer coordinates (n_0, n_1), with node (0,0) located in the upper left corner of the fault.

Line 1: Along-dip node range for the main fault zone. These are the lowest and highest n_0 values for nodes in the main fault zone.

Line 2: Along-strike node range for the main fault zone. These are the lowest and highest n_1 values for nodes in the main fault zone.

Line 3: Along-dip node range for the transition zone. These are the lowest and highest n_0 values for nodes in the transition zone.

Line 4: Along-strike node range for the transition zone. hese are the lowest and highest n_1 values for nodes in the transition zone.

Line 5: Along-dip node range for the stopping zone. These are the lowest and highest n_0 values for nodes in the stopping zone.

Line 6: Along-strike node range for the stopping zone. These are the lowest and highest n_1 values for nodes in the stopping zone.

In the example, the main fault zone measures 226 cells (= 16950) meters vertically and 320 cells (= 24000 meters) along strike. The transition zone measures 226 cells (= 16950) meters vertically and 480 cells (= 36000 meters) along strike. The stopping zone measures 226 cells (= 16950) meters vertically and 640 cells (= 48000 meters) along strike.

```
  0  226
160  480
  0  226
 80  560
  0  226
  0  640
```

The final line contains an integer, which is the number of bins used for constructing histograms of the stress distribution. They appear in the statistical information output files, and they are also used to compute medians of the stress distribution.

```
50
```

6. Input File Format for Phase 2, Friction Parameters and Stress Scaling

The input file supplies the parameters that the code needs to scale the stress field, set up the nucleation process, and insert the friction parameters into the output file. When you start the program, you must redirect standard input so the program reads parameters from the input file.

The input file is a text file, which must conform to Fortran input requirements. Each parameter can be an integer, real number, or character string. The program expects each line in the file to contain a certain number of parameters, as described below. Any text that appears on the line after the last expected parameter is treated as a comment, and is ignored. In the sample input files we adopted the convention that comments begin with a # character, however the program does not actually look for the # character. The input file may include blank lines, however, there is no provision for the input file to include a line that contains just a comment.

In this document, we show an example of an input file, accompanied by a description and explanation. The contents of the sample file are shown as **`blue courier text`**. The description and explanation is shown as ordinary black text. The example shown here is the sample input file tpvx05_fric_200-201.txt.

The first line of the file must contain the character string “run_fric01”. This commands the program to scale the stress distributions.

```
run_fric01
```

The next line contains two integers, which specify a range of realization numbers to be processed. In the example, the program processes realization numbers 200 through 201 inclusive. (It is not necessary to specify subrealization numbers, because the program determines them automatically.)

```
200  201
```

The following two character strings are used to construct the names of input files, which contain the raw stress distribution for each realization generated during phase 1. The file format is documented in section 8. These strings must be identical to the strings specified in the phase 1 input file. Each file name is constructed by inserting the realization and subrealization numbers in between the two character strings, which are used as a prefix and suffix respectively. In this example, there are twenty files named tpvx05_00200_000_raw.txt through tpvx05_00200_009_raw.txt. and tpvx05_00201_000_raw.txt through tpvx05_00201_009_raw.txt.

```
tpvx05  raw.txt
```

The following two character strings are used to construct the names of output files, which contain scaled stresses, in the same format as the raw stress files mentioned above. (For each node on the fault, the file contains one number, which is the ratio of shear stress divided by normal stress.) The file format is documented in section 8. Each file name is constructed by

inserting the realization and subrealization numbers in between the two character strings, which are used as a prefix and suffix respectively. In this example, there are twenty output files named tpvx05_00200_000_fstr.txt through tpvx05_00200_009_fstr.txt and tpvx05_00201_000_fstr.txt through tpvx05_00201_009_fstr.txt.

tpvx05 fstr.txt

The following two character strings are used to construct the names of output files, which contain scaled stresses, friction parameters, the nucleation process, and depth conditioning. **This is the file that is intended to be used as input to a dynamic rupture simulation.** The file format is documented in section 7. Each file name is constructed by inserting the realization and subrealization numbers in between the two character strings, which are used as a prefix and suffix respectively. In this example, there are twenty output files named tpvx05_00200_000_fric.txt through tpvx05_00200_009_fric.txt and tpvx05_00201_000_fric.txt through tpvx05_00201_009_fric.txt.

tpvx05 fric.txt

The following two character strings are used to construct the names of output files, which contain some statistical information about each generated realization. Each file name is constructed by inserting the realization and subrealization numbers in between the two character strings, which are used as a prefix and suffix respectively. In this example, there are twenty output files named tpvx05_00200_000_fout.txt through tpvx05_00200_009_fout.txt and tpvx05_00201_000_fout.txt through tpvx05_00201_009_fout.txt.

tpvx05 fout.txt

The following two character strings are used to construct the names of an output file, which contains a one-line summary of each generated realization. The file name is constructed by inserting the first realization number in between the two character strings, which are used as a prefix and suffix respectively. In this example, the output file is named tpvx05_00200_fsum.txt.

tpvx05 fsum.txt

The next sixteen lines contain a series of parameters, which specify how to scale the stresses. The method of scaling, and the meaning of these parameters, is described in section 12. The parameters are as follows:

Line 1: Target scale measure S_t.
Line 2: Target level measure L_t.
Line 3: Scale weight A_0^*.
Line 4: Scale weights A_1^* through A_5^*.
Line 5: Scale weights A_6^* through A_{10}^*.
Line 6: Scale weights A_{11}^* through A_{15}^*.
Line 7: Scale weights A_1 through A_5.
Line 8: Scale weights A_6 through A_{10}.

Line 9: Scale weights A_{11} through A_{15}.
Line 10: Level weight B_0^*.
Line 11: Level weights B_1^* through B_5^*.
Line 12: Level weights B_6^* through B_{10}^*.
Line 13: Level weights B_{11}^* through B_{15}^*.
Line 14: Level weights B_1 through B_5.
Line 15: Level weights B_6 through B_{10}.
Line 16: Level weights B_{11} through B_{15}.

In this example, the scale measure $S(\tau_i)$ is defined to be the mean stress in the main fault zone, minus the mean stress in the stopping zone, averaged over all twenty realizations that are being processed. The target scale S_t=0.23332 is set to a value that is the intended difference in stress between the main fault zone (where stress is high enough so ruptures propagate) and the stopping zone (where stress is low enough so ruptures stop). Recall that the "stress" τ referred to here is actually the ratio of shear stress divided by normal stress, so the target value should be considered to be in "friction coefficient space." See section 12 for an explanation of the scale measure, and section 3 for a diagram of fault zones.

Also, in this example, the level measure $L(\tau_i)$ is defined to be the mean value of the i-th realization stress, τ_i, in the stopping zone. The target level L_t=0.25634 is set to a value that is the intended level of stress in the stopping zone, chosen to be low enough to stop rupture propagation.

```
0.23332
0.25634

0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  1.0  0.0  0.0
0.0  0.0 -1.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0

0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0
0.0  0.0  1.0  0.0  0.0
```

The following lines specify the minimum allowed value of the scaled stress. The value is given as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. At each point on the fault surface, the result of the scaling operation selected previously is compared to this minimum value, and the stress is

increased if necessary to comply with the minimum. Note that the minimum value is given as a ratio of shear stress divided by normal stress, so the value is comparable to a friction coefficient. In this example, the minimum value is 0.0386 everywhere on the fault surface.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
2
    0.0e3   1.000   1.000
   19.5e3   1.000   1.000
2
2
    0.0e3   0.0386   0.0386
   48.0e3   0.0386   0.0386
```

The following lines specify the maximum allowed value of the scaled stress. The value is given as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. At each point on the fault surface, the result of the scaling operation selected previously is compared to this maximum value, and the stress is decreased if necessary to comply with the maximum. Note that the maximum value is given as a ratio of shear stress divided by normal stress, so the value is comparable to a friction coefficient. In this example, the maximum value is 0.7074 everywhere on the fault surface.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
2
    0.0e3   1.000   1.000
   19.5e3   1.000   1.000
2
2
    0.0e3   0.7074   0.7074
   48.0e3   0.7074   0.7074
```

The following integer selects the nucleation method that is built in to the output file. It must have the value 1, 2, or 3, as follows. The example below selects the time-drop method.

1 – Nucleate using the time-drop method. In a nucleation zone surrounding the hypocenter, the friction at each point drops down to the dynamic (sliding) friction, at a time that depends on distance from the hypocenter. The result is an expanding front of forced nucleation.

2 – Nucleate using the overstress method. In a nucleation zone surrounding the hypocenter, the initial stress at each point is set to an initial value that exceeds the static friction (yield stress), forcing nucleation to occur immediately at the start of the simulation.

3 – No nucleation. There is no nucleation process built in to the output file. The dynamic rupture simulation code presumably nucleates the rupture using some other method. Note: If you select this option, you still have to include all the nucleation and hypocenter parameters in the input file, even though their values are not used.

```
1
```

The following real number is the nucleation overstress parameter. It is given as a ratio of shear stress divided by normal stress, and so is comparable to a friction coefficient.

If the time-drop method is selected, then the initial shear stress within the nucleation zone is forced to be at least the dynamic (sliding) friction plus the overstress parameter value. The purpose of this adjustment is to ensure there are no negative or extremely low stress drops within the nucleation zone, which could prevent the nucleation process from working.

If the overstress method is selected, then the initial shear stress within the nucleation zone is forced to be at least the static friction (yield stress) plus the overstress parameter value.

```
0.011666
```

When the time-drop method is selected, the following two real numbers give the velocity of the forced rupture front, in meters per second, in the along-dip and along-strike directions respectively. In this example, the velocity is 2424 meters per second in both directions, yielding a circular rupture front. We recommend using values in the range of 0.5 to 0.7 times the shear-wave velocity.

If the time-drop method is not selected, this line still has to be included in the input file, even though the values of the two real numbers are not used.

```
2424.0   2424.0
```

The following two lines determine the size of the nucleation zone, which surrounds the hypocenter. If the first line contains the integer 1, then the second line contains the radius of the nucleation zone in meters. If the first line contains the integer 2, then the second line contains the radius of the nucleation zone expressed in units of the Day radius. (The Day radius is the radius at which it is energetically favorable for a circular crack to expand; see the open-file report for discussion.) In the example below, the radius of the nucleation zone equals 0.25 times the Day radius.

```
2
0.250
```

The following real number is the aspect ratio of the nucleation zone, which is defined to be the radius in the along-strike direction divided by the radius in the along-dip direction. In the example below, the aspect ratio is 1.0, which makes the nucleation zone circular. A value other than 1.0 would produce an elliptical nucleation zone.

```
1.000
```

The following real number is the shear modulus in the neighborhood of the nucleation zone, in Pascal. It is used for computing the Day radius. (Since you don't know the hypocenter location in advance, this should be an average or typical value of the shear modulus in the region where the hypocenter might be located.) In the example below, the shear modulus is 32.0 GPa.

```
32.0e9
```

The following real number is the nucleation zone taper radius. It is the distance from the hypocenter at which the program should begin to taper off the nucleation process, expressed as a multiple of the nucleation zone radius. In the example below, the taper radius is 0.8 of the nucleation zone radius, which means that the nucleation process is tapered off in the outer most 20 percent of the nucleation zone. A value of 1.0 would specify no tapering.

If the time-drop method is selected, tapering is done by cutting in half the velocity of the forced rupture front. If the overstress method is selected, tapering is done by linearly tapering off the amount of overstress.

```
0.800
```

The following two lines define the zone of reduced fracture energy, in which the critical slip distance D_c is reduced from its usual value. Reducing the critical slip distance near the hypocenter makes it possible to use a much smaller nucleation zone than is otherwise possible. Refer to the open-file report for discussion.

The first line contains two real numbers, which give the inner and outer radii of the zone of reduced fracture energy, expressed as a multiple of the nucleation zone radius. The second line gives the critical slip distance inside the inner radius, expressed as a multiple of its usual value. Between the two radii, the critical slip distance is linearly tapered up to its usual value.

In the example below, the inner radius is 0.4 times the nucleation zone radius, and the outer radius is 4.0 times the nucleation zone radius. Within the inner radius, the critical slip distance D_c is 0.1 times its usual value. Between the two radii, the D_c is linearly tapered from 0.1 times its usual value up to its usual value. (The usual value of D_c is defined later in the input file.)

If no zone of reduced fracture energy is desired, then use -1.0 and -1.0 on the first line, and use 1.0 on the second line.

```
0.400   4.000
0.100
```

The following six lines control hypocenter selection. You specify two rectangular areas on the fault surface, an inner rectangle that contains the eligible hypocenters, and an outer rectangle that that contains the eligible nucleation zones. A node on the fault surface is considered to be a candidate hypocenter if it lies within (or on the border of) the inner rectangle, and if its nucleation zone lies entirely within the outer rectangle. The program computes the average shear stress in the neighborhood of each candidate hypocenter, and selects the candidate that has the highest stress. The average shear stress in the neighborhood of a node is computed as the weighted average of the stress at nearby nodes, using a Gaussian weight function.

(The program aborts if it cannot find a hypocenter, that is, if there are no acceptable candidates. Typically, this means that the nucleation zone is too large to fit in the outer rectangle. This could be caused by a poor choice of parameters in this file, but it could also be caused by setting the correlation length Λ too small during phase 1. In the latter case, you need to re-run phase 1 with a larger value for Λ.)

The lines contain the following parameters:

Line 1: Two integers, giving the radius of the weight function, along-dip and along-strike respectively, measured in cells. In the example, the radius of 10 cells in each direction, which means the weight function occupies a square measuring 19 nodes by 19 nodes.

Line 2: Two real numbers, giving the standard deviation of the Gaussian weight function, in the along-dip and along-strike directions respectively, measured in cells. In the example, the standard deviation is 7 cells in each direction.

Line 3: Two integers, giving the along-dip node range for the rectangle of eligible hypocenter locations. These are the lowest and highest n_0 values for nodes that are eligible hypocenters.

Line 4: Two integers, giving the along-strike node range for the rectangle of eligible hypocenter locations. These are the lowest and highest n_1 values for nodes that are eligible hypocenters.

Line 5: Two integers, giving the along-dip node range for the rectangle of eligible nucleation zone locations. These are the lowest and highest n_0 values for nodes that are eligible to be withing the nucleation zone.

Line 6: Two integers, giving the along-dip node range for the rectangle of eligible nucleation zone locations. These are the lowest and highest n_1 values for nodes that are eligible to be withing the nucleation zone.

In the example below, the Gaussian weight function extends for 10 cells in each direction, which means it occupies a square measuring 19 nodes by 19 nodes. The standard deviation of the Gaussian weight function is 7 cells in each direction. Recalling that nodes in this example are spaced 75 meters apart, the rectangle of eligible hypocenters measures 5025 meters along-dip and 19950 meters along-strike, and is located 6000 meters below the earth's surface. The

rectangle of eligible nucleation zone locations measures 6975 meters along-dip and 24000 meters along-strike, and is located 5025 meters below the earth's surface.

```
10  10
7.0  7.0
 80  147
187  453
 67  160
160  480
```

In order to compute the Day radius, the program needs to know the average shear stress. The following two lines define a rectangular region on the fault surface. When computing the Day radius, the program uses the average shear stress within this rectangle.

Line 1: Along-dip node range for the rectangle. These are the lowest and highest n_0 values for nodes in the rectangle.

Line 1: Along-strike node range for the rectangle. These are the lowest and highest n_1 values for nodes in the rectangle.

In the example below, the rectangle measures 8025 meters along-dip and 24000 meters along-strike, and is located 3975 meters below the earth's surface (recalling that nodes are spaced 75 meters apart).

```
 53  160
160  480
```

The following two real numbers are multipliers that are applied to stress in the along-dip and along-strike directions respectively.

The thermodynamic method yields a single random shear stress value at each node on the fault surface. Dynamic rupture simulations require two components of shear stress at each node, an along-dip component and an along-strike component. The two components are computed by multiplying each of these two multipliers by the single random shear stress value. Here are commonly-used multiplier values:

0.0 1.0 → Right-lateral strike-slip.

0.0 −1.0 → Left-lateral strike-slip.

1.0 0.0 → Downward dip-slip.

−1.0 0.0 → Upward dip-slip.

$\sin\theta$ $\cos\theta$ → Rake angle θ.

The example below uses right-lateral strike-slip.

```
0.0  1.0
```

The following lines specify the initial normal stress σ_0, in Pascal, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. See section 4 for an explanation of friction parameters. In this example, σ_0 is 60.0 MPa everywhere on the fault surface.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
2
    0.0e3   60.0e6   60.0e6
   19.5e3   60.0e6   60.0e6
2
2
    0.0e3   1.000   1.000
   48.0e3   1.000   1.000
```

The following lines specify the static coefficient of friction μ_s, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. See section 4 for an explanation of friction parameters. In this example, μ_s is 0.667 everywhere on the fault surface.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
2
    0.0e3   0.677   0.677
   19.5e3   0.677   0.677
2
2
    0.0e3   1.000   1.000
   48.0e3   1.000   1.000
```

The following lines specify the dynamic coefficient of friction μ_d, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. See section 4 for an explanation of friction parameters. In this example, μ_d is 0.373 everywhere on the fault surface.

```
2
1.0  0.0  0.0
```

```
0.0  1.0  0.0
0.0
2
2
    0.0e3   0.373   0.373
   19.5e3   0.373   0.373
2
2
    0.0e3   1.000   1.000
   48.0e3   1.000   1.000
```

The following lines specify the critical slip distance D_c, in meters, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. See section 4 for an explanation of friction parameters. In this example, D_c is 0.400 meters everywhere on the fault surface.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
2
    0.0e3   0.400   0.400
   19.5e3   0.400   0.400
2
2
    0.0e3   1.000   1.000
   48.0e3   1.000   1.000
```

The following lines specify the cohesion C, in Pascal, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. See section 4 for an explanation of friction parameters. In this example, C is specified as

$$C(x_0, x_1) \equiv \begin{cases} (4000/3)(3000 - x_0), & \text{if } x_0 < 3000 \\ 0, & \text{if } 3000 \leq x_0 \end{cases}$$

In other words, the cohesion is 4.0 MPa at the earth's surface, and linearly tapers down to zero at a depth of 3 km.

```
2                       # Cohesion: value2d_vt_product = 2
1.0  0.0  0.0           # Linear transformation 0: mxx, mxy, bx
0.0  1.0  0.0           # Linear transformation 1: myx, myy, by
0.0                     # Value offset
2                       # value1d_vt_pwlinear = 2
3                       # Size of list 0
    0.0e3   4.00e6   4.00e6      # x0, v0_lo, v0_hi
    3.0e3   0.00e6   0.00e6      # x0, v0_lo, v0_hi
```

```
   19.5e3   0.00e6   0.00e6      # x0, v0_lo, v0_hi
2                   # value1d_vt_pwlinear = 2
2                   # Size of list 1
    0.0e3   1.000   1.000      # x1, v1_lo, v1_hi
   48.0e3   1.000   1.000      # x1, v1_lo, v1_hi
```

The following lines specify the depth conditioning function $\Omega(x_0, x_1)$, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. Refer to the open-file report for a discussion of depth conditioning. Depth conditioning is typically used to stop the rupture at the bottom of the fault surface. After all other scaling is applied, the stress field is modified according to the formula

$$\tau(x_0, x_1) \leftarrow \tau(x_0, x_1)\, \Omega(x_0, x_1)$$

By taking $\Omega(x_0, x_1)$ to be small near the bottom of the fault, you can produce low stresses to stop the rupture at the bottom of the fault. In the example, $\Omega(x_0, x_1)$ is specified as

$$\Omega(x_0, x_1) \equiv \begin{cases} 1.00, & \text{if } x_0 \leq 14000 \\ 0.80 + 0.0001(16000 - x_0), & \text{if } 14000 \leq x_0 \leq 16000 \\ 0.10 + 0.0002(19500 - x_0), & \text{if } 16000 \leq x_0 \leq 19500 \end{cases}$$

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
4
    0.0e3   1.000   1.000
   14.0e3   1.000   1.000
   16.0e3   0.800   0.800
   19.5e3   0.100   0.100
2
2
    0.0e3   1.000   1.000
   48.0e3   1.000   1.000
```

The following lines specify the minimum value of the strength excess coefficient, as a function of position (x_0, x_1) on the fault surface. The lines represent a function of two variables, in the manner described in section 11. The strength excess coefficient is defined to be $\mu_s - \tau_0/\sigma_0$, where μ_s is the static coefficient of friction, τ_0 is the initial shear stress, and σ_0 is the initial normal stress. See section 4 for an explanation of friction parameters. At any point on the fault where the strength excess coefficient is too small, the static coefficient of friction μ_s is automatically increased to comply with the minimum. This mechanism ensures that points on the fault do not immediately rupture at the beginning of a dynamic rupture simulation (except as part of the selected nucleation process). In this example, the minimum strength excess coefficient is 0.023332 everywhere on the fault surface.

```
2
1.0  0.0  0.0
0.0  1.0  0.0
0.0
2
2
    0.0e3   0.023332   0.023332
   19.5e3   0.023332   0.023332
2
2
    0.0e3   1.000   1.000
   48.0e3   1.000   1.000
```

The next seven lines specify the three rectangular zones on the fault surface, and the number of bins used for constructing histograms of the stress distribution. See the figure in section 3. These seven lines must be identical to the last seven lines in the input file for phase 1. Refer to section 5 for details.

```
  0  226
160  480
  0  226
 80  560
  0  226
  0  640
50
```

7. Output File Format

Each output file represents one realization of heterogeneous initial stress. It contains all the information needed to run a dynamic rupture simulation, not just the initial shear stresses, but also the friction parameters, initial normal stresses, hypocenter location, and nucleation method. This file is produced as output during phase 2 of the computation. **This file is intended to be used as the input to a dynamic rupture simulation**, and so it contains all the information needed to set up a dynamic rupture simulation. Refer to section 3 for a diagram of the fault surface, and the layout of nodes.

The following example shows the first seven lines of an output file. (Each long line in the file is printed here on three lines, to make it fit on the page. It may help to look at an actual output file.)

```
200     3
640   260   4.800000E+04   1.950000E+04
364   117   2.730000E+04   8.775000E+03   1.127366E+03   1.127366E+03
  0     0   0.000000E+00   0.000000E+00   6.000000E+07   2.509795E+07
            0.000000E+00    4.182992E-01   0.000000E+00    6.770000E-01
            3.730000E-01    4.000000E-01   4.000000E+06    1.000000E+09
  1     0   7.500000E+01   0.000000E+00   6.000000E+07   2.523141E+07
            0.000000E+00    4.205235E-01   0.000000E+00    6.770000E-01
            3.730000E-01    4.000000E-01   4.000000E+06    1.000000E+09
  2     0   1.500000E+02   0.000000E+00   6.000000E+07   2.524198E+07
            0.000000E+00    4.206997E-01   0.000000E+00    6.770000E-01
            3.730000E-01    4.000000E-01   4.000000E+06    1.000000E+09
  3     0   2.250000E+02   0.000000E+00   6.000000E+07   2.512238E+07
            0.000000E+00    4.187063E-01   0.000000E+00    6.770000E-01
            3.730000E-01    4.000000E-01   4.000000E+06    1.000000E+09
```

The first three lines are a header, and the remaining lines contain data. The header contains information about the realization, but is not actually needed to run the simulation. In your code, you may wish to skip over the three-line header, and just read the data.

Line 1: Identifies the random number seed used to generate the initial stresses. The fields are:

- Realization number, an integer.
- Sub-realization number, an integer.

Line 2: Contains the total size of the fault surface. The fields are:

- Number of cells along-strike (this is one less than the number of nodes along-strike). The example shows a fault with 640 cells, or 641 nodes, along-strike.

- Number of cells down-dip (this is one less than the number of nodes down-dip). The example shows a fault with 260 cells, or 261 nodes, along-dip.

- Size of fault surface along-strike, in meters. In the example, it is 48000. The spacing between nodes along-strike can be computed by dividing this value by the number of cells; in the example it is 48000/640 = 75 meters.

- Size of fault surface down-dip, in meters. In the example, it is 19500. The spacing between nodes along-dip can be computed by dividing this value by the number of cells; in the example it is 19500/260 = 75 meters.

Line 3: Contains the hypocenter location. The fields are:

- Hypocenter location along-strike, measured in cells (counting from 0 at the left edge of the fault surface). It is an integer. The value is -1 if there is no nucleation information in the file.

- Hypocenter location down-dip, measured in cells (counting from 0 at the earth's surface). It is an integer. The value is -1 if there is no nucleation information in the file.

- Hypocenter location along-strike, in meters, from the left edge of the fault surface.

- Hypocenter location down-dip, in meters, from the earth's surface.

- Radius of the zone of forced rupture, along-strike, in meters.

- Radius of the zone of forced rupture, along-dip, in meters.

Remaining lines give the initial stresses and friction parameters for each node on the fault surface. In the example, there would be 641×261 = 167301 of these lines. The fields on each line are:

- Node location along-strike, measured in cells (counting from 0 at the left edge of the fault surface). This is coordinate n_1. It is an integer.

- Node location down-dip, measured in cells (counting from 0 at the earth's surface). This is coordinate n_0. It is an integer.

- Node location along-strike, in meters, from the left edge of the fault surface.

- Node location down-dip, in meters, from the earth's surface.

- Initial normal stress, in Pascal. Positive values denote compression. This is σ_0. In the example, this equals 60.0 MPa at all nodes.

- Initial shear stress along-strike, in Pascal. The sign convention is that positive values denote stresses that tend to cause right-lateral slip. In the example, this is the initial shear stress τ_0.

- Initial shear stress along-dip, in Pascal. The sign convention is that positive values denote stresses that tend to cause downward slip (the far side of the fault moving downward relative to the near side). In the example, this is zero at all nodes.

- Ratio of initial shear stress along-strike divided by initial normal stress. In the example, this is τ_0/σ_0.

- Ratio of initial shear stress along-dip divided by initial normal stress. In the example, this is zero.

- Static coefficient of friction. This is μ_s. In the example this is 0.677, except that the value is increased at points where the initial shear stress is very high.

- Dynamic coefficient of friction. This is μ_d. In the example, this is 0.373.

- Slip-weakening critical distance, in meters. This is D_c. In the example, this is 0.40 m, except in the vicinity of the hypocenter where it is reduced.

- Cohesion, in Pascal. This is C. In the example, the cohesion is zero except in the upper 3 km where it tapers up to 4.0 MPa at the earth's surface.

- Time of forced rupture, in seconds. This is T. For nodes outside the zone of forced rupture, this value is 1.0E+9, indicating that the node does not undergo forced rupture. For nodes inside the zone, this is the time at which the friction is forced to drop down to the dynamic sliding friction (e.g., by reducing the static coefficient of friction to equal the dynamic coefficient of friction). At the hypocenter, the time is zero. If there is no nucleation information in the file, then the value is 1.0E+9 at all nodes.

Nodes are listed in a specific order. The node in the upper left corner of the fault surface is listed first. Next come the remaining nodes at the earth's surface, listed left to right. Next come the nodes one cell below the earth's surface, listed left to right. And so on. At the end of the file come the nodes at the bottom of the fault surface, listed left to right, with the node in the lower right corner of the fault surface appearing last.

8. Raw Stress File Format

This file format is used for the raw stress file, which is written out during phase 1 of the computation, and then read back in during phase 2 of the computation. This file format is also used for the scaled stress file that is written out during phase 2. The file contains nothing more than a single stress value for each node on the fault surface. This file can be useful for making plots of the random stress distribution. Refer to section 3 for a diagram of the fault surface, and the layout of nodes.

The following example shows the first four lines and last eleven lines of a raw stress file. (Each long line in the file is printed here on two lines, to make it fit on the page. It may help to look at an actual file.)

```
   261        641   -2.403750E+04    2.403750E+04
                      0.000000E+00    1.957500E+04
4.182992E-01    4.205234E-01    4.206996E-01    4.187063E-01
4.156998E-01    4.154451E-01    4.143325E-01    4.098320E-01
4.006195E-01    3.927872E-01    3.884817E-01    3.855406E-01

  .   .   .

3.832221E-01    3.902896E-01    3.870572E-01    3.758669E-01
3.615526E-01         (this line is called "Line N" in the description below)
   200          3
7.500000E+01    7.500000E+01
     1
   364        117
1.127365E+03    1.127365E+03
1.906261E+12
3.860000E-02    6.913075E-01    3.951816E-01
                 1.188972E-01    4.127041E-01
2.634738E-01    6.913075E-01    4.787260E-01
                 7.272321E-02    4.823385E-01
3.860000E-02    5.152494E-01    2.563704E-01
                 9.178212E-02    2.443202E-01
```

The first line is a header, and the last nine lines are a trailer. The remaining lines contain the stress values, one value per node.

Line 1: Contains the total size of the fault surface. (It is presented in a slightly unusual way, which indicates the size that should be used to construct a plot of the stresses.) The fields are:

- Number of nodes along-dip. The example shows a fault with 261 nodes along-dip.

- Number of nodes along-strike. The example shows a fault with 641 nodes along-strike.

- The value $-h_1 N_1/2$, where h_1 is the spacing between nodes along-strike, and N_1 is the number of nodes along-strike. In this example, h_1=75 meters and N_1=641, so the value is -24037.5 meters.

- The value $h_1 N_1/2$, where h_1 is the spacing between nodes along-strike, and N_1 is the number of nodes along-strike. In this example, h_1=75 meters and N_1=641, so the value is 24037.5 meters.

- The value 0.0.

- The value $h_0 N_0$, where h_0 is the spacing between nodes along-dip, and N_0 is the number of nodes along-dip. In this example, h_0=75 meters and N_0=261, so the value is 19575 meters.

Lines 2 through N: Contain the stress values. These may be raw stresses, or scaled stresses. There is one real number for each node. Nodes are listed row-by-row. The node in the upper left corner of the fault surface is listed first. Next come the remaining nodes at the earth's surface, listed left to right. Next come the nodes one cell below the earth's surface, listed left to right. And so on. At the end come the nodes at the bottom of the fault surface, listed left to right, with the node in the lower right corner of the fault surface appearing last. Four nodes are listed on each line, except that each row of nodes begins at the start of a new line. In this example, there are 641×261 = 167301 nodes listed.

Line N+1: Identifies the random number seed used to generate the initial stresses. The fields are:

- Realization number, an integer.

- Sub-realization number, an integer.

Line N+2: Contains the spacing between nodes. The fields are:

- Spacing between nodes, along-strike, in meters.

- Spacing between nodes, along-dip, in meters.

Line N+3: Contains a single integer, which is always 1.

Line N+4: Identifies the node where the hypocenter is located. The fields are:

- Hypocenter location along-strike, measured in cells (counting from 0 at the left edge of the fault surface). It is an integer. The value is -1 if the hypocenter location is unknown or

unspecified.

- Hypocenter location along-dip, measured in cells (counting from 0 at the earth's surface). It is an integer. The value is -1 if the hypocenter location is unknown or unspecified.

Line N+5: Contains the size of the nucleation zone (the zone of forced rupture that surrounds the hypocenter). The fields are:

- Radius of the nucleation zone, along-strike, in meters. The value is 0.0 if the hypocenter location is unknown or unspecified.

- Radius of the nucleation zone, along-dip, in meters. The value is 0.0 if the hypocenter location is unknown or unspecified.

Line N+6: Contains a single real number, which is a measure of how well the selection template Q matches the random stress distribution.

Line N+7: Contains some statistical properties of the stress distribution, over the entire fault surface. The fields are:

- Minimum stress value.

- Maximum stress value.

- Mean stress value.

- Standard deviation of the stress distribution.

- Median stress value.

Line N+8: Contains some statistical properties of the stress distribution, over the main fault zone. (See section 3 for a diagram of fault zones.) The fields are:

- Minimum stress value.

- Maximum stress value.

- Mean stress value.

- Standard deviation of the stress distribution.

- Median stress value.

Line N+9: Contains some statistical properties of the stress distribution, over the stopping zone. (See section 3 for a diagram of fault zones.) The fields are:

- Minimum stress value.
- Maximum stress value.
- Mean stress value.
- Standard deviation of the stress distribution.
- Median stress value.

9. Random Number Generation

To initialize the random number generator, it is necessary to supply a random seed. The code requires a random seed text file, which contains eleven integers on each line. The realization number selects one line in the file, and the eleven integers on that line are used to seed the random number generator. Lines in the random seed file are numbered beginning with zero, so realization number 0 uses the seeds from the first line of the file, realization number 1 uses the seeds from the second line of the file, and so on.

We have supplied a sample random seed file, which contains the first 250 lines of the Rand table of random digits. The full table can be found on the Rand website at: http://www.rand.org/pubs/monograph_reports/MR1418.html. You can, of course, supply your own random seed file.

The code consumes an enormous quantity of random numbers, and so it is very important to use a high-quality random number generator. We have included the Mersenne Twister random number generator, which is a very high-quality generator suitable for use in Monte Carlo simulations. Information can be found on the Mersenne Twister website at: http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html.

10. Representing a Function of One Variable

In the input files, it is necessary to specify some functions of one variable. These are usually used to construct functions of two variables, as described in section 11. The code supports the following functions of one variable:

- A constant function.
- A piecewise linear function.

It would be straightforward to modify the code to add more functions, if that were needed.

The following subsections show how different functions of one variable are represented in the input file. As before, `blue courier text` shows the contents of the input file, and ordinary black text is the explanation

Constant function

The following represents a constant function whose value is:

$$f(x) = v \qquad \text{for all } x$$

This function is represented in the input file as shown below. The integer 1 on the first line indicates that this is a constant function. The second line contains the constant value v.

```
1
v
```

Piecewise linear function

The following represents a piecewise linear function. We are given a list of n control points x_i, $i = 1, \ldots, n$, which must satisfy $x_1 < x_2 < \cdots < x_n$. We are also given corresponding values v_i and w_i. The piecewise linear function is then defined to be:

$$f(x) = \begin{cases} v_1, & \text{if } x < x_1 \\ ((x - x_i)v_{i+1} + (x_{i+1} - x)w_i)/(x_{i+1} - x_i), & \text{if } x_i \le x < x_{i+1} \\ w_n, & \text{if } x_n \le x \end{cases}$$

The middle line above is the standard formula for linear interpolation between the points (x_i, w_i) and (x_{i+1}, v_{i+1}). Notice that if $v_i \neq w_i$ then the function is discontinuous at $x = x_i$.

This function is represented in the input file as shown below. The integer 2 on the first line indicates that this is a piecewise linear function. The second line contains the integer n which

gives the number of control points, and must satisfy $n \geq 1$. The succeeding n lines contain the control points x_i and their corresponding values v_i and w_i, for $i = 1, \ldots, n$.

```
2
n
x1   v1   w1
x2   v2   w2
.  .  .
xn   vn   wn
```

11. Representing a Function of Two Variables

In the input files, it is necessary to specify some functions of two variables. For instance, the selection template and the friction parameters are given as functions of position (x_0, x_1) on the fault surface. The code supports the following functions of two variables:

- A constant function.
- A product function.
- A radial function.

It would be straightforward to modify the code to add more functions, if that were needed.

Each function of two variables has a built-in affine transformation, which allows the function to be rotated, translated, magnified, and sheared.

The following subsections show how different functions of two variables are represented in the input file. As before, `blue courier text` shows the contents of the input file, and ordinary black text is the explanation

Constant function

The following represents a constant function whose value $f(x, y)$ is defined as:

$$x' = m_{xx}x + m_{xx}y + b_x$$

$$y' = m_{yx}x + m_{yy}y + b_y$$

$$f(x, y) = v \qquad \text{for all } x, y$$

(The values of x' and y' are computed, but are not used in the constant function.)

This function is represented in the input file as shown below. The integer 1 on the first line indicates that this is a constant function. The second line contains the three coefficients used to calculate x'. The third line contains the three coefficients used to calculate y'. The fourth line contains the constant value v.

```
1
mxx   mxy   bx
myx   myy   by
v
```

Product function

The following represents a product function whose value $f(x, y)$ is defined as:

$$x' = m_{xx}x + m_{xx}y + b_x$$

$$y' = m_{yx}x + m_{yy}y + b_y$$

$$f(x, y) = v + g(x')h(y')$$

where g and h are arbitrary functions of one variable.

Several examples of product functions can be found in sections 5 and 6.

This function is represented in the input file as shown below. The integer 2 on the first line indicates that this is a product function. The second line contains the three coefficients used to calculate x'. The third line contains the three coefficients used to calculate y'. The fourth line contains the value v. Succeeding lines contain the function g of one variable, followed by the function h of one variable. The functions g and h each occupy multiple lines, and can be any of the functions of one variable described in section 10.

```
2
mxx   mxy   bx
myx   myy   by
v
g
h
```

Radial function

The following represents a radial function whose value $f(x, y)$ is defined as:

$$x' = m_{xx}x + m_{xx}y + b_x$$

$$y' = m_{yx}x + m_{yy}y + b_y$$

$$f(x, y) = g\left(\sqrt{x'^2 + y'^2}\right)$$

where g is an arbitrary function of one variable.

This function is represented in the input file as shown below. The integer 3 on the first line indicates that this is a radial function. The second line contains the three coefficients used to calculate x'. The third line contains the three coefficients used to calculate y'. Succeeding lines

contain the function g of one variable. The function g occupies multiple lines, and can be any of the functions of one variable described in section 10.

```
3
mxx   mxy   bx
myx   myy   by
g
```

12. Scaling the Stress Field

In phase 1 of the calculation, the program generates a raw stress field for each realization. In phase 2 of the computation, the raw stress fields are scaled to be consistent with the friction law. This section describes how you control the scaling process.

For the i-th realization, in phase 1 the program generates a raw stress value $\tau_i(n_0, n_1)$ at each node location (n_0, n_1). Refer to section 3 for a diagram of node locations. Recall from the open-file report that the raw stress τ actually represents the ratio of shear stress divided by normal stress, and so τ should be regarded as living in "friction coefficient space." In phase 2, the program applies a linear scaling to each realization of random stress, of the form

$$\tau_i'(n_0, n_1) \leftarrow a_i \cdot \tau_i(n_0, n_1) + b_i$$

The goal is for the scaled stress τ_i' to bear an appropriate relation to the static and dynamic coefficients of friction, μ_s and μ_d.Our concern here is how to select the constants a_i and b_i that are used for scaling.

Suppose that we have some measures of the *scale* and *level* of the raw stress fields. Specifically, suppose we have a *scale* measure $S(\tau_i)$ which measures the amplitude of the raw stress field τ_i. For example, we might take $S(\tau_i)$ to be the standard deviation of $\tau_i(n_0, n_1)$ over some part of the fault surface, or we might take $S(\tau_i)$ to be the difference between the maximum and minimum values of $\tau_i(n_0, n_1)$. Also, suppose we have a *level* measure $L(\tau_i)$ which measures the typical value of the raw stress field τ_i. For example, we might take $L(\tau_i)$ to be the mean or median value of $\tau_i(n_0, n_1)$ over some part of the fault surface.

Now suppose we select *target* values for the scale and level, which we denote S_t and L_t. The target values are the scale and level that a stress field ought to have, so that it is consistent with the friction law. For example, we might choose the target level L_t to be somewhere between μ_s and μ_d, and we might choose the target scale S_t to be some fraction of $\mu_s - \mu_d$. Having chosen the target scale and level, the scaling parameters a_i and b_i are easily computed as:

$$a_i = S_t / S(\tau_i)$$

$$b_i = L_t - a_i L(\tau_i)$$

Given these formulas, all that is necessary is to specify the desired scale and level measures, and target values. The code provides a flexible, but somewhat cumbersome, method for specifying the measures.

The code permits the scale and level measures $S(\tau_i)$ and $L(\tau_i)$ to depend not only on the stress field τ_i of the i-th realization, but also on the entire set of stress fields that are being processed. This permits the measures to vary randomly among realizations. For example, if $S(\tau_i)$ is taken to be the standard deviation of τ_i, then every scaled stress field τ_i' has standard deviation exactly equal to S_t. But if $S(\tau_i)$ is taken to be the average standard deviation of all the realizations under

consideration, then all the a_i are equal and the standard deviation of τ_i' can vary from one realization to another, with an average standard deviation of S_t.

To construct the scale and level measures, the code computes a set of fifteen statistical measures for each realization, as shown in the following table. Refer to section 3 for a diagram of the zones on the fault surface.

M_1^i	Minimum value of $\tau_i(n_0, n_1)$ over the entire fault surface.
M_2^i	Maximum value of $\tau_i(n_0, n_1)$ over the entire fault surface.
M_3^i	Mean value of $\tau_i(n_0, n_1)$ over the entire fault surface.
M_4^i	Standard deviation of $\tau_i(n_0, n_1)$ over the entire fault surface.
M_5^i	Median value of $\tau_i(n_0, n_1)$ over the entire fault surface.
M_6^i	Minimum value of $\tau_i(n_0, n_1)$ over the main fault zone.
M_7^i	Maximum value of $\tau_i(n_0, n_1)$ over the main fault zone.
M_8^i	Mean value of $\tau_i(n_0, n_1)$ over the main fault zone.
M_9^i	Standard deviation of $\tau_i(n_0, n_1)$ over the main fault zone.
M_{10}^i	Median value of $\tau_i(n_0, n_1)$ over the main fault zone.
M_{11}^i	Minimum value of $\tau_i(n_0, n_1)$ over the stopping zone.
M_{12}^i	Maximum value of $\tau_i(n_0, n_1)$ over the stopping zone.
M_{13}^i	Mean value of $\tau_i(n_0, n_1)$ over the stopping zone.
M_{14}^i	Standard deviation of $\tau_i(n_0, n_1)$ over the stopping zone.
M_{15}^i	Median value of $\tau_i(n_0, n_1)$ over the stopping zone.

For each $k = 1, \ldots, 15$, let M_k^* denote the average value of M_k^i over all realizations i under consideration. So for example, M_5^* is the average of the medians of the raw stress fields over the entire fault.

To specify the scale measure $S(\tau_i)$, you specify thirty-one constants, A_1 through A_{15} and A_0^* through A_{15}^*, and then

$$S(\tau_i) \equiv A_0^* + \sum_{k=1}^{15} (A_k M_k^i + A_k^* M_k^*)$$

Likewise, to specify the level measure $L(\tau_i)$, you specify thirty-one constants, B_1 through B_{15} and B_0^* through B_{15}^*, and then

$$L(\tau_i) \equiv B_0^* + \sum_{k=1}^{15} (B_k M_k^i + B_k^* M_k^*)$$

We recommend choosing $A_1 = \cdots = A_{15} = 0$ because then all the scaling parameters a_i are equal.

The open-file report contains additional discussion of stress scaling.