

# TPV24 and TPV25 Branched-Fault Benchmarks

January 23, 2013

These 3D benchmarks use a branching fault in a linear elastic half-space. We are doing two benchmarks, as shown in the following table.

<b>Benchmarks</b>			
<i>Benchmark</i>	<i>Dimension</i>	<i>Rupture Type</i>	<i>Material Properties</i>
TPV24	3D	Right-lateral, releasing branch.	Linear elastic.
TPV25	3D	Left-lateral, restraining branch.	Linear elastic.

We request that you run these benchmarks using two resolutions: 100 meter resolution, and 50 meter resolution.

The fault geometry is the same as in benchmarks TPV14-15 and TPV18-21. However, the hypocenter and station locations have been changed.

The material properties are the same as in benchmark TPV5 and several other benchmarks.

These benchmarks use linear elastic material properties, but they have a very similar design to benchmarks that use plastic material properties.

We are introducing some changes from earlier benchmarks:

- The fault boundary condition is that slip goes to zero at the border of the fault. So, a node which lies precisely on the border of a fault should *not* be permitted to slip.
- Nucleation is done using a smoothed forced rupture. The forced rupture is applied over an interval of 0.5 seconds (instead of being applied instantaneously as in earlier benchmarks), and the forced rupture velocity varies with distance from the hypocenter.

## Contents

Part 1: Considerations for Plastic Benchmarks .....	3
Part 2: Description of the 3D Benchmarks .....	4
Fault Geometry .....	4
Material Properties .....	5
Initial Stress Tensor .....	5
Friction Parameters and Nucleation.....	8
Running Time, Node Spacing, and Results .....	9
Part 3: Junction Point Behavior .....	10
Part 4: Linear Slip-Weakening Friction .....	11
Part 5: Nucleation .....	13
Part 6: Methods for Implementing Gravity and Boundary Traction .....	14
Method 1 — Using Stress Change.....	14
Method 2 — Using Explicit Gravity and Boundary Traction .....	14
Part 7: On-Fault Stations, and Time-Series File Format.....	17
Part 8: Off-Fault Stations, and Time-Series File Format .....	23
Part 9: Contour-Plot File Format .....	28

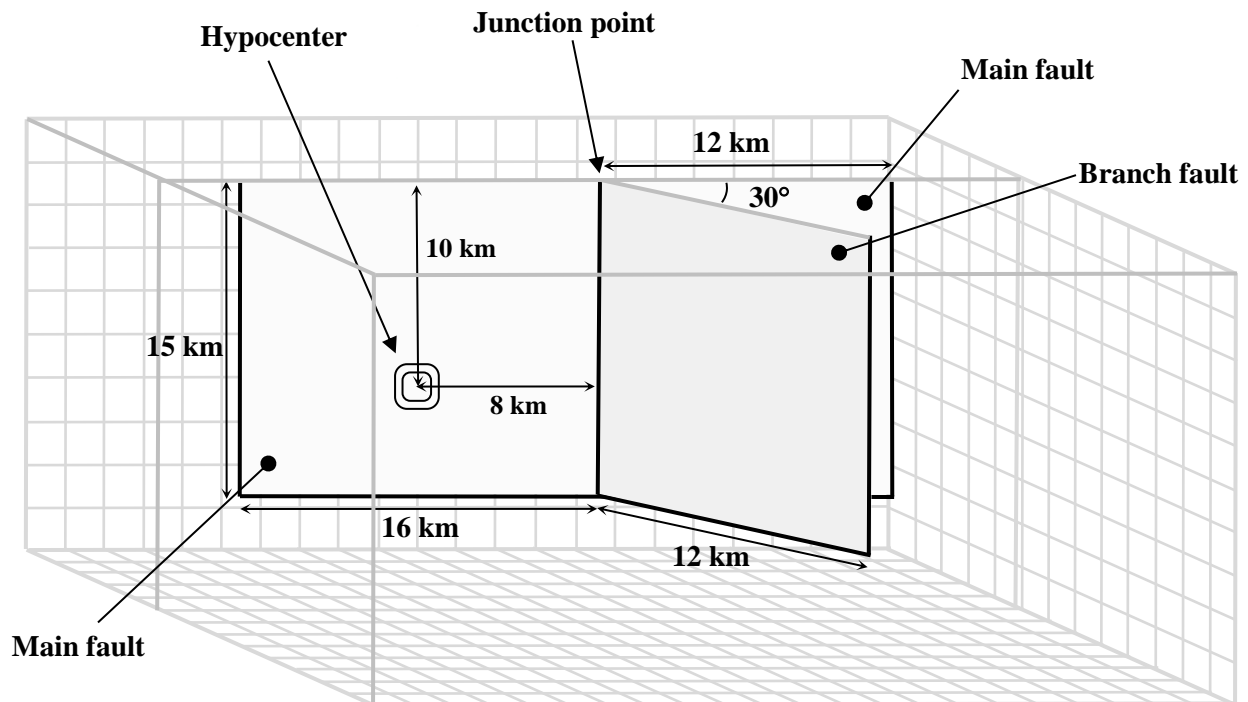
## ***Part 1: Considerations for Plastic Benchmarks***

These benchmarks use linear elastic material properties, but are designed in the same way that plastic benchmarks would be designed. The following list highlights some special considerations to keep in mind.

- **The initial stress tensor is specified everywhere throughout the entire model volume.** Benchmarks with linear elasticity only need to specify initial shear and normal stresses on the fault surfaces. In TPV24 and TPV25, the initial shear and normal stresses on the faults are implied by resolving the initial stress tensor onto the fault surfaces.
- **Gravity must be accounted for.** The initial stress increases with depth, so gravitational acceleration is required to balance the gradient in the initial stress tensor.
- **You may need to apply traction forces to the boundary of the model volume.** The initial stresses exert traction forces on the boundary of the model volume. Depending on your implementation, you may need to balance them with externally-applied tractions.
- **Fluid pressure is included in the model.** Fluid pressure is hydrostatic, with water table at the earth's surface. The fluid pressure enters into the plastic constitutive law, and the friction law.
- **Nucleation is done using forced rupture.** Benchmarks with linear elasticity can simply impose higher shear stress near the hypocenter, but that is not possible with plasticity because it would be inconsistent with the initial stress tensor. In TPV24 and TPV25, we nucleate by using a zone of forced rupture surrounding the hypocenter. A detailed description is provided.

**TPV24 and TPV25 continue to use a linear slip-weakening friction law.** As with TPV14-15 and TPV18-21, we use vertical faults that branch along-strike, with a 30-degree branch angle, in a uniform half-space.

## Part 2: Description of the 3D Benchmarks



### Fault Geometry

The model volume is a half-space.

There are two faults, called the **main fault** and the **branch fault**. The two faults are vertical, planar, strike-slip faults. The faults reach the earth's surface. In TPV24, they are right-lateral faults. In TPV25, they are left-lateral faults.

The main fault is a rectangle measuring 28000 m along-strike and 15000 m deep. The branch fault is a rectangle measuring 12000 m along-strike and 15000 m deep. There is a junction point. It is located 12000 m from the right edge of the main fault, and the main fault passes through it. The branch fault makes an angle of 30 degrees to the main fault. The branch fault ends at the junction point.

The hypocenter is centered along-strike at a depth of 10 km in the left side of the main fault. That is, the hypocenter is 8000 m from the junction point, and 10000 m deep.

**Slip goes to zero at the border of a fault. So, a node which lies precisely on the border of a fault should not be permitted to slip. This is a change from earlier benchmarks.**

**The main fault runs continuously through the junction point. The branch fault ends at the junction point. On the branch fault, the slip vector is required to go to zero at the junction point, just like it does at any other border of a fault.** This is discussed in detail later.

## Material Properties

In TPV24 and TPV25, the entire model volume is a linear elastic material, with the following parameters.

$$\text{Density } \rho = 2670 \text{ kg/m}^3$$

$$\text{Shear-wave velocity } V_s = 3464 \text{ m/s}$$

$$\text{Pressure-wave velocity } V_p = 6000 \text{ m/s}$$

Note that these are the same material properties used in benchmark TPV5 and several other benchmarks.

## Initial Stress Tensor

TPV24 and TPV25 are linear elastic benchmarks, but they are designed in the same way as plastic benchmarks. In an inelastic calculation, the initial stress tensor needs to be specified throughout the medium. The total (time-varying) stress, which equals the initial stress plus the stress change, is used in the plastic yield calculation. The (time-varying) normal and shear stresses acting on the fault surface are implied by the stress tensor in the adjacent rock.

For TPV24 and TPV25, the initial stress tensor varies only with depth. The components of the stress tensor are defined in the following table.

<b>Stress Tensor Components</b>	
<i>Component</i>	<i>Definition</i>
$\sigma_{11}$	Compressive stress in the vertical direction. Negative values denote compression.
$\sigma_{22}$	Compressive stress in the direction parallel to the main fault. Negative values denote compression.
$\sigma_{33}$	Compressive stress in the direction perpendicular to the main fault. Negative values denote compression. This equals the negative of the total normal stress on the main fault.
$\sigma_{23}$	Shear stress in the horizontal plane. Positive values denote right-lateral shear stress on the main fault. This equals the shear stress on the main fault.
$\sigma_{13}$	Shear stress in a vertical plane perpendicular to the main fault.
$\sigma_{12}$	Shear stress in a vertical plane parallel to the main fault.

The stress tensor components can vary with time during the simulation.

Fluid pressure  $P_f$  is hydrostatic with water table at the surface. It varies with depth, but is constant in time.

For the 3D benchmarks, the fluid pressure and the initial values of the stress tensor are:

$$P_f = (1000 \text{ kg/m}^3)(9.8 \text{ m/s}^2)(\text{depth in meters})$$

$$\sigma_{11} = -(2670 \text{ kg/m}^3)(9.8 \text{ m/s}^2)(\text{depth in meters})$$

$$\sigma_{22} = \begin{cases} b_{22} (\sigma_{11} + P_f) - P_f, & \text{if depth} \leq 15600 \text{ m} \\ \sigma_{11}, & \text{if depth} > 15600 \text{ m} \end{cases}$$

$$\sigma_{33} = \begin{cases} b_{33} (\sigma_{11} + P_f) - P_f, & \text{if depth} \leq 15600 \text{ m} \\ \sigma_{11}, & \text{if depth} > 15600 \text{ m} \end{cases}$$

$$\sigma_{23} = \begin{cases} b_{23} (\sigma_{11} + P_f), & \text{if depth} \leq 15600 \text{ m} \\ 0, & \text{if depth} > 15600 \text{ m} \end{cases}$$

$$\sigma_{13} = 0$$

$$\sigma_{12} = 0$$

The coefficients  $b_{22}$ ,  $b_{33}$ , and  $b_{23}$  are given in the following table.

Initial Stress Tensor Coefficients		
Coefficient	Value for TPV24	Value for TPV25
$b_{22}$	0.926793	1.119338
$b_{33}$	1.073206	0.880661
$b_{23}$	-0.169029	0.138704

**These coefficient values are the only difference between the right-lateral benchmark (TPV24) and the left-lateral benchmark (TPV25).**

Note that we have chosen the gravitational acceleration  $g$  to be *exactly*  $9.8 \text{ m/s}^2$ . Do not substitute a “more accurate” value of  $g$ .

For both benchmarks, the initial axis of intermediate compressive stress is vertical, and its initial value is lithostatic. The initial axes of maximum and minimum compressive stress are horizontal, and their average value is lithostatic.

Note that we defined the stress tensor components  $\sigma_{ij}$  to be the **total** stress tensor, which includes both elastic stress and fluid pressure. To obtain the **effective** stress tensor, which includes only elastic stress, you need to add the fluid pressure  $P_f$  to the diagonal components. This can be shown as:

$$\text{total stress tensor} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} \end{pmatrix}$$

$$\text{effective stress tensor} = \begin{pmatrix} \sigma_{11} + P_f & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} + P_f & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} + P_f \end{pmatrix}$$

## Friction Parameters and Nucleation

We use a linear slip-weakening friction law, which has the following six parameters.

Friction Parameters		
<i>Symbol</i>	<i>Parameter</i>	<i>Unit</i>
$\mu_s$	Static coefficient of friction.	Dimensionless
$\mu_d$	Dynamic coefficient of friction.	Dimensionless
$d_0$	Slip-weakening critical distance.	Meter
$C_0$	Frictional cohesion.	Pascal
$T$	Time of forced rupture.	Second
$t_0$	Forced rupture decay time	Second

The operation of the slip-weakening friction law is described in detail later, in part 4.

The friction parameter values are as follows, where  $r$  denotes distance to the hypocenter:

$$\mu_s = 0.18$$

$$\mu_d = 0.12$$

$$d_0 = 0.30 \text{ m}$$

$$C_0 = \begin{cases} 0.30 \text{ MPa} + (0.000675 \text{ MPa/m})(4000 \text{ m} - \text{depth}), & \text{if depth} \leq 4000 \text{ m} \\ 0.30 \text{ MPa}, & \text{if depth} \geq 4000 \text{ m} \end{cases}$$

$$r_{\text{crit}} = 4000 \text{ m}$$

$$T = \begin{cases} \frac{r}{0.7 V_S} + \frac{0.081 r_{\text{crit}}}{0.7 V_S} \left( \frac{1}{1 - (r/r_{\text{crit}})^2} - 1 \right), & \text{if } r < r_{\text{crit}} \\ 1.0\text{E}+9, & \text{if } r \geq r_{\text{crit}} \end{cases}$$

$$t_0 = 0.50 \text{ s}$$

The variation in the value of  $T$  near the hypocenter causes the rupture to nucleate, in a manner that is described later, in part 5.

Note that the frictional cohesion  $C_0$  is 3.00 MPa at the earth's surface. It is 0.30 MPa at depths greater than 4000 m, and its value is linearly tapered in the uppermost 4000 m.



## Running Time, Node Spacing, and Results

Run the model for times from **0.0 to 12.0 seconds after nucleation**.

Please submit results for two resolutions:

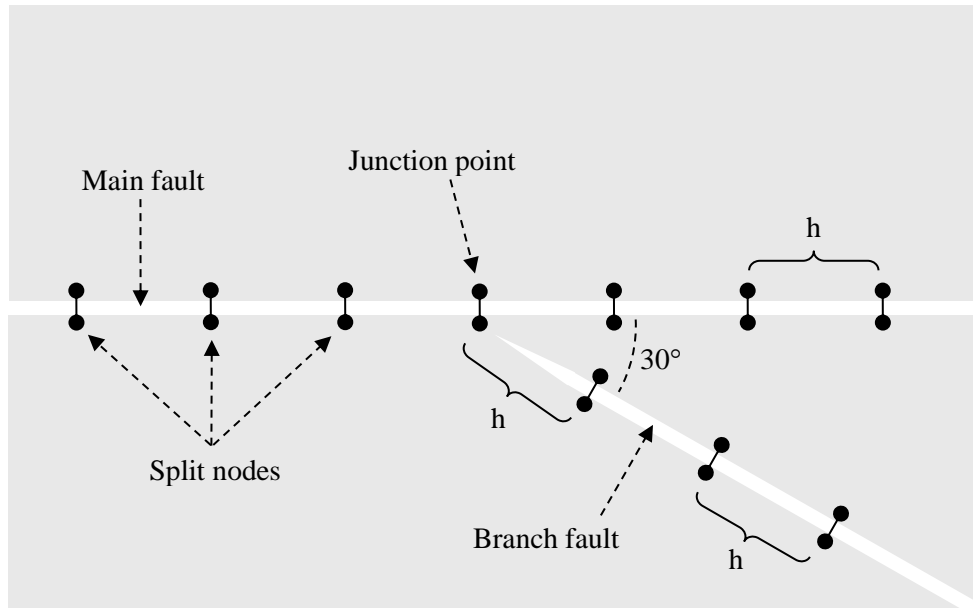
- Using **100 m node spacing** on the fault planes.
- Using **50 m node spacing** on the fault planes.

Use the “**Change Version**” function on the website to submit your two sets of results. If you are unable to run the simulation with **50 m node spacing**, then it is **OK to submit just 100 m results**.

The requested output files are:

- **On-fault time-series files**, which give slips, slip rates, and stresses for each on-fault station at each time step. These files are described in part 7.
- **Off-fault time-series files**, which give displacements and velocities for each off-fault station at each time step. These files are described in part 8.
- **Contour-plot files** which, for each node on one of the two faults, gives the time at which the slip rate first changes from 0 to greater than 0.001 m/s. There are two contour-plot files, one for the main fault and one for the branch fault. These files are described in part 9.

### Part 3: Junction Point Behavior



The physical behavior of the junction point is defined as follows: **On the branch fault, the slip vector goes to zero at the junction point.**

The figure is an example of how this behavior can be implemented by a finite-element code. Other types of codes may implement this behavior in other ways. The figure shows the main and branch faults near the junction point. The faults are implemented using split nodes, which in the figure are spaced at intervals  $h$  along the faults. For a simulation with 100 m resolution,  $h = 100$  m. For a simulation with 50 m resolution,  $h = 50$  m. Each split node consists of a pair of nodes on opposite sides of the fault. Fault slip is achieved by allowing the two halves of a split node to move relative to each other.

**The main fault runs continuously through the junction point.** At the junction point, the main fault has an ordinary split node.

**The branch fault ends at the junction point.** The first split node on the branch fault is located a distance  $h$  from the junction point, measured parallel to the strike of the branch fault as shown in the figure. (So, the first split node on the branch fault is located at  $h/2$  perpendicular distance from the main fault.) The branch fault does not have a split node at the junction point.

## Part 4: Linear Slip-Weakening Friction

Benchmarks TPV24 and TPV25 use linear slip-weakening friction, with frictional cohesion and forced rupture. This friction law has the following parameters and variables:

Friction Parameters		
<i>Symbol</i>	<i>Parameter</i>	<i>Unit</i>
$\mu_s$	Static coefficient of friction.	Dimensionless
$\mu_d$	Dynamic coefficient of friction.	Dimensionless
$d_0$	Slip-weakening critical distance.	Meter
$C_0$	Frictional cohesion.	Pascal
$T$	Time of forced rupture.	Second
$t_0$	Forced rupture decay time	Second

Friction Variables		
<i>Symbol</i>	<i>Parameter</i>	<i>Unit</i>
$\sigma_n$	Total normal stress acting on the fault, taken to be positive in compression.	Pascal
$\tau$	Shear stress acting on the fault.	Pascal
$P_f$	Fluid pressure (which is assumed to be constant in time).	Pascal
$\sigma_n - P_f$	Effective normal stress acting on the fault, which is positive in compression.	Pascal

When the fault is sliding, the shear stress  $\tau$  at a given point on the fault is given by:

$$\tau = C_0 + \mu \max(0, \sigma_n - P_f)$$

The time-varying coefficient of friction  $\mu$  is given by the following formulas, where  $D$  is the total distance the node has slipped, and  $t$  is the time since the start of the simulation:

$$\mu = \mu_s + (\mu_d - \mu_s) \max(f_1, f_2)$$

$$f_1 = \begin{cases} D/d_0, & \text{if } D < d_0 \\ 1, & \text{if } D \geq d_0 \end{cases}$$

$$f_2 = \begin{cases} 0, & \text{if } t < T \\ (t - T)/t_0, & \text{if } T \leq t < T + t_0 \\ 1, & \text{if } t \geq T + t_0 \end{cases}$$

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, as described in part 5. At time  $t = T$ , the coefficient of friction begins to decrease, reaching its final dynamic value at time  $t = T + t_0$ . So, the node is forced to begin sliding somewhere in the interval  $T < t < T + t_0$ . Notice that because of the expression  $\max(f_1, f_2)$ , forcing has no effect if friction has already weakened due to the accumulated slip. If a given node does not undergo forced rupture, then the value of  $T$  is 1.0E9.

**Tension on the fault:** If you encounter tension on the fault, you should **treat tension on the fault the same as if the effective normal stress equals zero**. This is shown in the above formulas by the expression  $\max(0, \sigma_n - P_f)$ .

You should **constrain the motion of the node so that the fault cannot open (that is, only permit sliding parallel to the fault), even when the fault is in tension**. During the time the fault is in tension, continue to accumulate the slip distance  $D$  as usual.

**In your time-series files, where you are asked to report “normal stress,” you should report the effective normal stress** which is defined to be  $\sigma_n - P_f$ .

## Part 5: Nucleation

Nucleation is performed by forcing the fault to rupture, within a circular zone surrounding the hypocenter. Forced rupture is achieved by artificially reducing the friction coefficient, beginning at a specified time  $T$ . The parameter  $t_0$  specifies how long it takes for the friction coefficient to be artificially reduced from its static value to its dynamic value. So, the friction coefficient reaches its dynamic value at time  $T + t_0$ . We reduce the friction coefficient gradually, over an interval of time, in order to smooth the nucleation process and reduce unwanted oscillations.

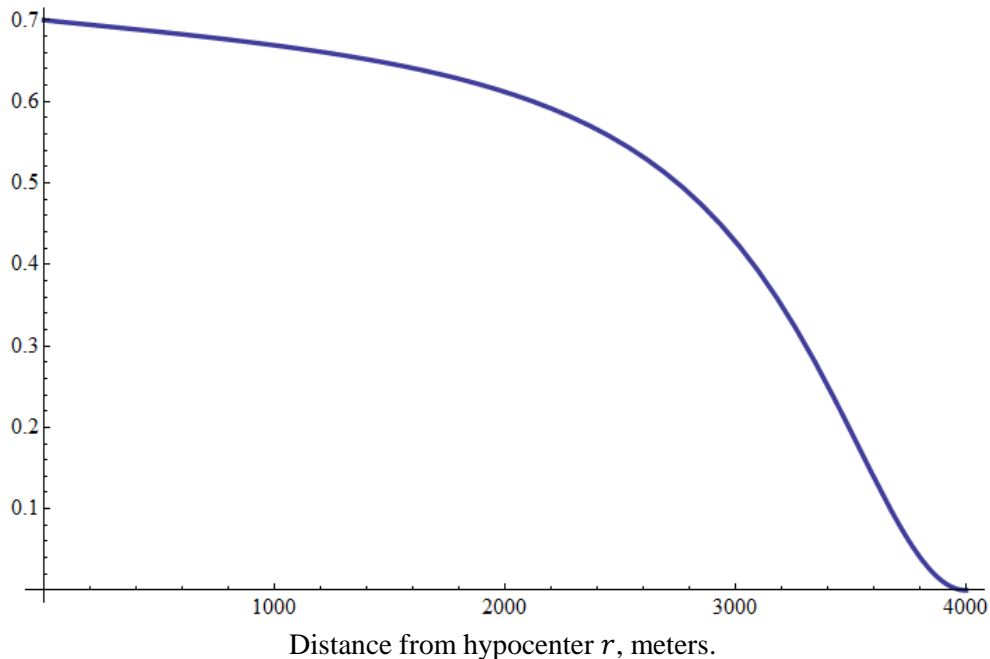
Recall that we selected  $t_0 = 0.5$  s, and that the time of forced rupture is defined to be

$$T = \begin{cases} \frac{r}{0.7 V_S} + \frac{0.081 r_{\text{crit}}}{0.7 V_S} \left( \frac{1}{1 - (r/r_{\text{crit}})^2} - 1 \right), & \text{if } r < r_{\text{crit}} \\ 1.0\text{E}+9, & \text{if } r \geq r_{\text{crit}} \end{cases}$$

where  $r$  is distance from the hypocenter, and where  $r_{\text{crit}} = 4000$  m is the radius of the nucleation zone. At the hypocenter,  $T = 0$ . The value of  $T$  then increases with distance from the hypocenter, which creates an expanding circular region of forced rupture.

The time  $T$  is computed so that the forced rupture expands at a variable speed. Near the hypocenter, the forced rupture expands at a speed of  $0.7 V_S$ . The speed decreases with increasing distance from the hypocenter, finally reaching a speed of zero at a distance of 4000 m from the hypocenter. The variable speed allows for a smooth transition between forced rupture and spontaneous rupture, because the spontaneous rupture should gradually overtake the ever-slowing forced rupture. The graph below shows the forced rupture speed  $V_R \equiv dr/dT$  as a function of distance from the hypocenter.

Speed of forced rupture  $V_R/V_S$ .



## ***Part 6: Methods for Implementing Gravity and Boundary Traction***

In an inelastic calculation, the initial stress needs to be specified throughout the medium. The total stress, which equals initial stress plus stress change, is used in the yield calculation. This creates two challenges.

- Because the initial stress varies with depth, a gravitational force is required to balance the gradient of the initial stress tensor.
- The initial stress tensor exerts traction forces on the boundary of the mesh. If the code permits the mesh boundaries to move, then external tractions must be applied to balance these forces.

There are two basic methods for dealing with gravity and boundary tractions, described below. You may choose either method.

### **Method 1 — Using Stress Change**

The stress tensor that is calculated and stored in the code is the change of stress from some initial state. The initial state is assumed to be in static equilibrium. With this method, the gravity and boundary tractions do not appear explicitly in the calculation, because as far as the code is concerned, all the initial stresses and forces are equal to zero.

In an elastic calculation, the initial state does not need to be known. In an inelastic calculation, the initial stress tensor must be specified and stored within each element (it varies only with depth in this case). Performing the plastic yield calculation then requires three steps:

1. Add the stress change to the initial stress, obtaining the total stress tensor.
2. Perform the plastic yield calculation (for example, with the return map algorithm), obtaining a new total stress tensor.
3. Subtract the initial stress from the new total stress tensor, obtaining a new stress change.

Because the code starts off with all stresses and forces equal to zero, you must “manually” specify the initial shear stress and initial effective normal stress at each node on the fault surface. To calculate the frictional sliding on the fault, you must (a) calculate the tractions on the fault surface implied by the change in stress, and then (b) add the initial shear stress and initial effective normal stress, to obtain the total shear stress and total effective normal stress which appear in the friction law.

### **Method 2 — Using Explicit Gravity and Boundary Traction**

The stress tensor that is calculated and stored in the code is the total stress tensor. The stress tensor throughout the medium must be initialized to a state that is in static equilibrium. The equilibrium stress state might come from a static calculation. In this case, because it depends only on depth, the initial stress state is simple enough to specify directly.

The acceleration of gravity must be added explicitly to the equations of motion. The gradient of the initial stress tensor is balanced by gravitational forces. For TPV24 and TPV25, you should set the gravitational acceleration  $g$  to be exactly  $9.8 \text{ m/s}^2$ .

If your code allows the mesh boundaries to move, then you must also apply external tractions to the mesh boundary, to balance the tractions caused by the initial stress. You need to calculate and store the force exerted on each boundary node by the initial stress conditions (including the gravitational force). Then, during the dynamic calculation, apply the negative of the stored force to each node lying in the boundary, thereby eliminating unwanted motion of the boundary. There are two ways to calculate the forces:

1. Perform a preliminary static calculation.
2. Analytically calculate the forces, in a manner that depends on the details of your code.

Because of approximation errors in the equations of motion, there may be small velocities throughout the mesh before real signals arrive. They are caused by small residual forces that exist because the nodal forces calculated from the initial stress tensor may not precisely cancel out the calculated gravitational forces. This effect is generally small enough to be ignored.

(Remark: If you do a preliminary static calculation to obtain boundary tractions, you can at the same time calculate the residual forces. Then, during the dynamic calculation, you can cancel out both the boundary tractions and the residual forces, thereby eliminating the small velocities. Just be careful that you don't also cancel out the traction forces acting on the fault surface!)

The plastic yield calculation has the total stress tensor available, so that the initial stress tensor does not need to be stored.

The code automatically calculates the total shear stress and total normal stress acting on the fault surface. So, it is not necessary to “manually” specify initial stresses acting on the fault surface.

However, the frictional sliding calculation also requires the fluid pressure, so that effective normal stress can be computed. There are two ways you could obtain the fluid pressure at each node on the fault surface. You may use either technique.

1. Calculate the fluid pressure by applying the formula

$$P_f = (1000 \text{ kg/m}^3)(9.8 \text{ m/s}^2)(\text{depth in meters})$$

This should work well for fault nodes not located on the earth's surface. However, special considerations apply to a fault node located at the earth's surface. For most codes it is incorrect to assign a fluid pressure of zero. We recommend that the fluid pressure be set as follows: (a) If your code has an initial stress tensor that is uniform within each element, then a fault node at the earth's surface should be assigned a fluid pressure corresponding to a depth of 1/2 element. (b) If

your code has an initial stress tensor that varies linearly with depth inside each element, then a fault node at the earth's surface should be assigned a fluid pressure corresponding to a depth of 1/3 element.

2. Perform a preliminary static calculation to obtain the traction exerted on the fault surface by fluid pressure. Set up the static calculation to use density and initial stress as

$$\rho = 1000 \text{ kg/m}^3$$
$$\sigma_{11} = \sigma_{22} = \sigma_{33} = -(1000 \text{ kg/m}^3)(9.8 \text{ m/s}^2)(\text{depth in meters})$$

The static calculation should also include the gravitational force. You can numerically calculate the normal stress exerted at each node on the fault surface by fluid pressure, and store the calculated normal stress. Then, during the dynamic calculation, you use the stored value as the fluid pressure.

(Because of numerical approximation errors, the static calculation may also yield a small shear stress acting on the fault surface. If you wish, you may store the calculated shear stress, and deduct it from the total shear stress (using vector subtraction) during the dynamic calculation.)

(Remark: Some codes may prefer to calculate and store the effective stress tensor at each point of the medium, rather than the total stress tensor. This does not fundamentally change the method, although some details are changed. In particular, because resolving the effective stress tensor onto the fault surface yields effective normal stress, it is not necessary to separately calculate the fluid pressure on the fault surface.)



## Part 7: On-Fault Stations, and Time-Series File Format

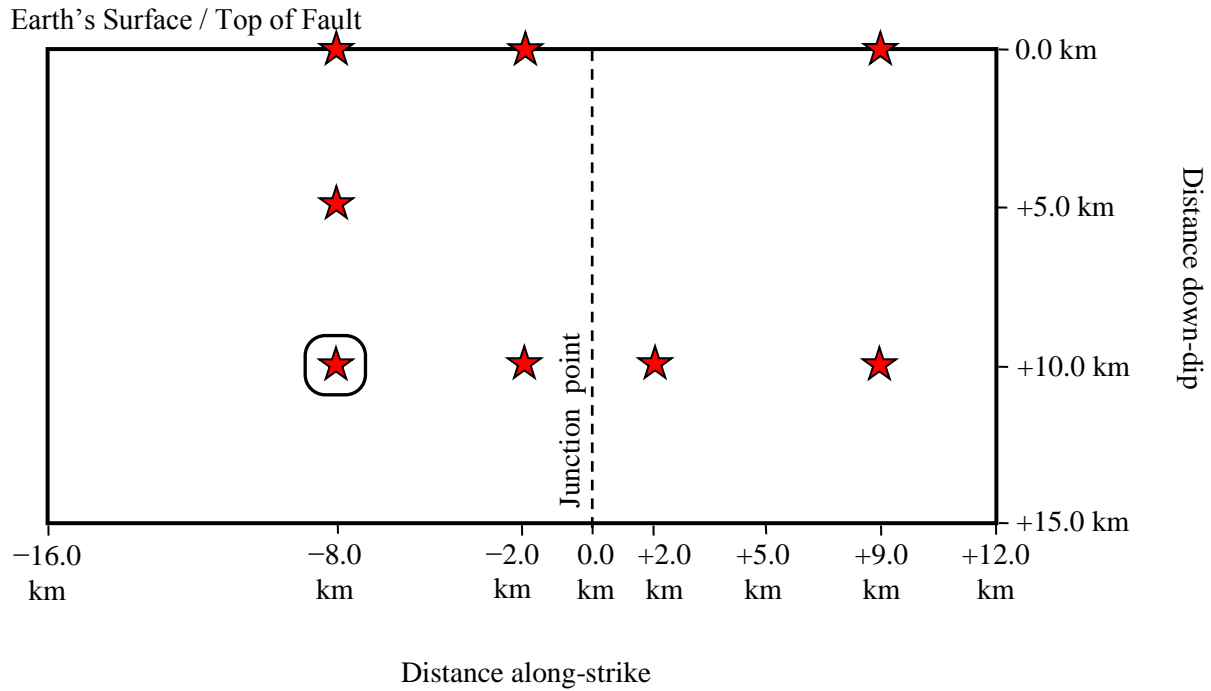
The 3D benchmarks use 8 stations on the main fault and 6 stations on the branch fault, which are listed below. Diagrams of station locations are given following the table. You need to supply one time-series file for each station.

On-Fault Stations, for TPV24 and TPV25	
Station Name	Location
faultst-080dp000	On main fault, -8.0 km along strike, 0 km down-dip.
faultst-020dp000	On main fault, -2.0 km along strike, 0 km down-dip.
faultst090dp000	On main fault, 9.0 km along strike, 0 km down-dip.
faultst-080dp050	On main fault, -8.0 km along strike, 5.0 km down-dip.
faultst-080dp100	On main fault, -8.0 km along strike, 10.0 km down-dip (hypocenter).
faultst-020dp100	On main fault, -2.0 km along strike, 10.0 km down-dip.
faultst020dp100	On main fault, 2.0 km along strike, 10.0 km down-dip.
faultst090dp100	On main fault, 9.0 km along strike, 10.0 km down-dip.
branchst020dp000	On branch fault, 2.0 km along strike, 0 km down-dip.
branchst090dp000	On branch fault, 9.0 km along strike, 0 km down-dip.
branchst020dp050	On branch fault, 2.0 km along strike, 5.0 km down-dip.
branchst010dp100	On branch fault, 1.0 km along strike, 10.0 km down-dip.
branchst020dp100	On branch fault, 2.0 km along strike, 10.0 km down-dip.
branchst090dp100	On branch fault, 9.0 km along strike, 10.0 km down-dip.

If you do not have a node at the location of a station, there are two options: (1) you can move the station to the nearest node, or (2) you can interpolate the data values from nodes near the station location.

**Note:** Location along-strike is measured relative to the junction point. Positive locations are to the right of the junction point. On the branch fault, distance is measured from the junction point parallel to the strike of the branch fault.

## Station Locations on the Main Fault



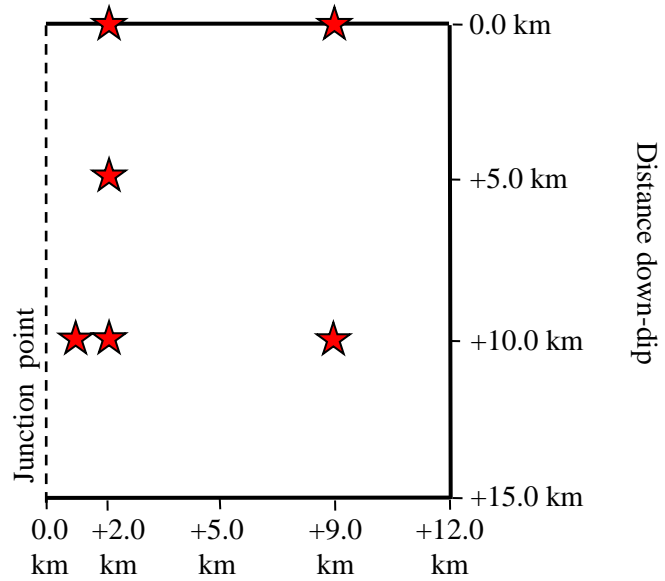
There are 8 stations:

- -8.0 km, -2.0 km, and +9.0 km along-strike, and 0 km down-dip distance.
- -8.0 km along-strike, and 5.0 km down-dip distance.
- -8.0 km, -2.0 km, +2.0 km, and +9.0 km along-strike, and 10.0 km down-dip distance.

Note that location along-strike is measured relative to the junction point. Positive locations are to the right of the junction point.

## Station Locations on the Branch Fault

Earth's Surface / Top of Fault



Distance along-strike

There are 6 stations:

- +2.0 km and +9.0 km along-strike, and 0 km down-dip distance.
- +2.0 km along-strike, and 5.0 km down-dip distance.
- +1.0 km, +2.0 km, and +9.0 km along-strike, and 10.0 km down-dip distance.

Note that location along-strike is measured relative to the junction point. Positive locations are to the right of the junction point.

Each time series file is an ASCII file that contains 8 data fields, as follows.

<b>On-Fault Time Series Data Fields for TPV24 and TPV25</b>	
<b>Field Name</b>	<b>Description, Units, and Sign Convention</b>
t	Time (s).
h-slip	Horizontal slip along-strike (m). Sign convention: Positive means <b>right lateral</b> slip.
h-slip-rate	Horizontal slip rate along-strike (m/s). Sign convention: Positive means <b>right lateral</b> motion.
h-shear-stress	Horizontal shear stress along-strike (MPa). Sign convention: Positive means shear stress that tends to cause <b>right-lateral</b> slip.
v-slip	Vertical along-dip slip (m). Sign convention: Positive means <b>downward slip</b> (that is, the far side of the fault moving downward relative to the near side of the fault).
v-slip-rate	Vertical along-dip slip rate (m/s). Sign convention: Positive means <b>downward motion</b> (that is, the far side of the fault moving downward relative to the near side of the fault).
v-shear-stress	Vertical along-dip shear stress (MPa). Sign convention: Positive means shear stress that tends to cause <b>downward slip</b> (that is, the far side of the fault moving downward relative to the near side of the fault).
n-stress	Normal stress (MPa). Note: You should report <b>effective normal stress</b> , which is defined to be total normal stress minus fluid pressure. Sign convention: Positive means <b>extension</b> .

The **near side** of a fault is in the front of the diagram.

The **far side** of a fault is in the back of the diagram.

The on-fault time series file consists of three sections, as follows.

<b>On-Fault Time Series File Format for TPV24 and TPV25</b>	
<b>File Section</b>	<b>Description</b>
File Header	<p>A series of lines, each beginning with a # symbol, that gives the following information:</p> <ul style="list-style-type: none"> <li>• Benchmark problem (TPV24 or TPV25)</li> <li>• Author</li> <li>• Date</li> <li>• Code</li> <li>• Code version</li> <li>• Node spacing or element size</li> <li>• Time step</li> <li>• Number of time steps in file</li> <li>• Station location</li> <li>• Descriptions of data columns (7 lines)</li> <li>• Anything else you think is relevant</li> </ul>
Field List	<p>A <b>single line</b>, which lists the names of the 8 data fields, in column order, separated by spaces. It should be:</p> <pre>t h-slip h-slip-rate h-shear-stress v-slip v-slip-rate v-shear-stress n-stress</pre> <p>(all on one line). The server examines this line to check that your file contains the correct data fields.</p>
Time History	<p>A series of lines. Each line contains 8 numbers, which give the data values for a single time step. The lines must appear in order of increasing time.</p> <p><b>C/C++ users:</b> For all data fields except the time, we recommend using 14.6E or 14.6e floating-point format. For the time field, we recommend using 20.12E or 20.12e format (but see the note on the next page).</p> <p><b>Fortran users:</b> For all data fields except the time, we recommend using E15.7 or 1PE15.6 floating-point format. For the time field, we recommend using E21.13 or 1PE21.12 format (but see the note on the next page).</p> <p>The server accepts most common numeric formats. If the server cannot understand your file, you will see an error message when you attempt to upload the file.</p>

**Note:** We recommend higher precision for the time field so the server can tell that your time steps are all equal. (If the server thinks your time steps are not all equal, it will refuse to apply digital filters to your data.) If you use a “simple” time step value like 0.01 seconds or 0.005 seconds, then there is no need for higher precision, and you can write the time using the same precision as all the other data fields. When you upload a file, the server will warn you if it thinks your time steps are not all equal.

Here is an example of an on-fault time-series file. This is an invented file, not real modeling data.

```
# Example on-fault time-series file.
#
# This is the file header:
# problem=TPV24
# author=A.Modeler
# date=2013/01/23
# code=MyCode
# code_version=3.7
# element_size=100 m
# time_step=0.005
# num_time_steps=2400
# location= on main fault, 9 km along strike, 7.5km down-dip
# Column #1 = Time (s)
# Column #2 = horizontal slip (m)
# Column #3 = horizontal slip rate (m/s)
# Column #4 = horizontal shear stress (MPa)
# Column #5 = vertical slip (m)
# Column #6 = vertical slip rate (m/s)
# Column #7 = vertical shear stress (MPa)
# Column #8 = normal stress (MPa)
#
# The line below lists the names of the data fields:
t h-slip h-slip-rate h-shear-stress v-slip v-slip-rate v-shear-stress n-stress
#
# Here is the time-series data.
# There should be 8 numbers on each line, but this page is not wide enough
# to show 8 numbers on a line, so we only show the first five.
0.000000E+00  0.000000E+00  0.000000E+00  7.000000E+01  0.000000E+00  ...
5.000000E-03  0.000000E+00  0.000000E+00  7.104040E+01  0.000000E+00  ...
1.000000E-02  0.000000E+00  0.000000E+00  7.239080E+01  0.000000E+00  ...
1.500000E-02  0.000000E+00  0.000000E+00  7.349000E+01  0.000000E+00  ...
2.000000E-02  0.000000E+00  0.000000E+00  7.440870E+01  0.000000E+00  ...
2.500000E-02  0.000000E+00  0.000000E+00  7.598240E+01  0.000000E+00  ...
# ... and so on.
```

## Part 8: Off-Fault Stations, and Time-Series File Format

The 3D benchmarks use the 8 off-fault stations listed below. All stations are at the earth's surface.

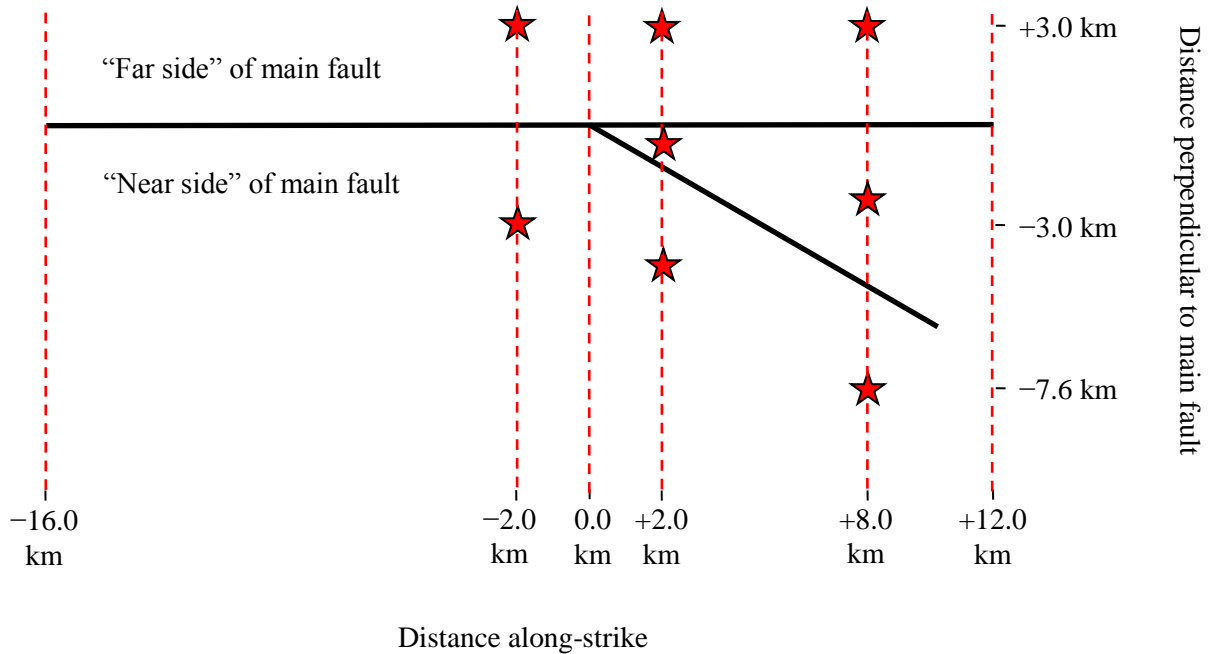
Refer to the next page for a diagram of station locations. You need to supply one time-series file for each station.

Off-Fault Stations for TPV24 and TPV25	
Station Name	Location
body030st-020dp000	3.0 km off main fault (far side), -2.0 km along strike, 0 km depth.
body-030st-020dp000	-3.0 km off main fault (near side), -2.0 km along strike, 0 km depth.
body030st020dp000	3.0 km off main fault (far side), 2.0 km along strike, 0 km depth.
body-006st020dp000	-0.6 km off main fault (near side), 2.0 km along strike, 0 km depth.
body-042st020dp000	-4.2 km off main fault (near side), 2.0 km along strike, 0 km depth.
body030st080dp000	3.0 km off main fault (far side), 8.0 km along strike, 0 km depth.
body-023st080dp000	-2.3 km off main fault (near side), 8.0 km along strike, 0 km depth.
body-076st080dp000	-7.6 km off main fault (near side), 8.0 km along strike, 0 km depth.

In the station names, the first number is the horizontal perpendicular distance from the station to the main fault. A positive number means that the station is located on the **far side** of the main fault.

If you do not have a node at the location of a station, there are two options: (1) you can move the station to the nearest node, or (2) you can interpolate the data values from nodes near the station location.

## Off-Fault Station Locations



The diagram shows the earth's surface, looking downwards.

There are 8 stations at the earth's surface:

- -2.0 km along strike, 0 km depth, and +3.0 km, -3.0 km perpendicular distance from the main fault trace.
- +2.0 km along strike, 0 km depth, and +3.0 km, -0.6 km, -4.2 km perpendicular distance from the main fault trace.
- +8.0 km along strike, 0 km depth, and +3.0 km, -2.3 km, -7.6 km perpendicular distance from the main fault trace.

The **near side** of the main fault is in the front of the diagram.

The **far side** of the main fault is in the back of the diagram.

**Positive** perpendicular distance from the main fault means that the station is on the **far side**.



Each time series file is an ASCII file that contains 7 data fields, as follows.

<b>Off-Fault Time Series Data Fields for TPV24 and TPV25</b>	
<b>Field Name</b>	<b>Description, Units, and Sign Convention</b>
t	Time (s).
h-disp	Horizontal displacement, parallel to the fault strike (m). Sign convention: Positive means displacement <b>to the right</b> relative to the station's initial position.
h-vel	Horizontal velocity, parallel to the fault strike (m/s). Sign convention: Positive means motion <b>to the right</b> .
v-disp	Vertical displacement (m). Sign convention: Positive means displacement <b>downward</b> relative to the station's initial position.
v-vel	Vertical velocity (m/s). Sign convention: Positive means motion <b>downward</b> .
n-disp	Horizontal displacement, perpendicular to the fault strike (m). Sign convention: Positive means displacement <b>away from the viewer, into the paper</b> (that is, away from near side of the main fault and toward the far side of the fault) relative to the station's initial position.
n-vel	Horizontal velocity, perpendicular to the fault strike (m/s). Sign convention: Positive means motion <b>away from the viewer, into the paper</b> (that is, away from near side of the main fault and toward the far side of the fault).

The **near side** of the main fault is in the front of the diagram.

The **far side** of the main fault is in the back of the diagram.

The off-fault time series file consists of three sections, as follows.

<b>Off-Fault Time Series File Format for TPV24 and TPV25</b>	
<b>File Section</b>	<b>Description</b>
File Header	<p>A series of lines, each beginning with a # symbol, that gives the following information:</p> <ul style="list-style-type: none"> <li>• Benchmark problem (TPV24 or TPV25)</li> <li>• Author</li> <li>• Date</li> <li>• Code</li> <li>• Code version</li> <li>• Node spacing or element size</li> <li>• Time step</li> <li>• Number of time steps in file</li> <li>• Station location</li> <li>• Descriptions of data columns (7 lines)</li> <li>• Anything else you think is relevant</li> </ul>
Field List	<p>A <b>single line</b>, which lists the names of the 7 data fields, in column order, separated by spaces. It should be:</p> <pre>t h-disp h-vel v-disp v-vel n-disp n-vel</pre> <p>(all on one line). The server examines this line to check that your file contains the correct data fields.</p>
Time History	<p>A series of lines. Each line contains 7 numbers, which give the data values for a single time step. The lines must appear in order of increasing time.</p> <p><b>C/C++ users:</b> For all data fields except the time, we recommend using 14.6E or 14.6e floating-point format. For the time field, we recommend using 20.12E or 20.12e format (but see the note on the next page).</p> <p><b>Fortran users:</b> For all data fields except the time, we recommend using E15.7 or 1PE15.6 floating-point format. For the time field, we recommend using E21.13 or 1PE21.12 format (but see the note on the next page).</p> <p>The server accepts most common numeric formats. If the server cannot understand your file, you will see an error message when you attempt to upload the file.</p>

**Note:** We recommend higher precision for the time field so the server can tell that your time steps are all equal. (If the server thinks your time steps are not all equal, it will refuse to apply digital filters to your data.) If you use a “simple” time step value like 0.01 seconds or 0.005 seconds, then there is no need for higher precision, and you can write the time using the same precision as all the other data fields. When you upload a file, the server will warn you if it thinks your time steps are not all equal.

Here is an example of an off-fault time-series file. This is an invented file, not real modeling data.

```
# Example off-fault time-series file.
#
# This is the file header:
# problem=TPV24
# author=A.Modeler
# date=2013/01/23
# code=MyCode
# code_version=3.7
# element_size=100 m
# time_step=0.005
# num_time_steps=2400
# location= 3.0 km off fault, 5 km along strike, 0.0km depth
# Column #1 = Time (s)
# Column #2 = horizontal displacement (m)
# Column #3 = horizontal velocity (m/s)
# Column #4 = vertical displacement (m)
# Column #5 = vertical velocity (m/s)
# Column #6 = normal displacement (m)
# Column #7 = normal velocity (m/s)
#
# The line below lists the names of the data fields:
t h-disp h-vel v-disp v-vel n-disp n-vel
#
# Here is the time-series data.
# There should be 7 numbers on each line, but this page is not wide enough
# to show 7 numbers on a line, so we only show the first five.
0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  0.000000E+00  ...
5.000000E-03  -2.077270E-85  -2.575055E-83  -2.922774E-86  -3.623018E-84  ...
1.000000E-02  -1.622118E-82  -2.005817E-80  -1.387778E-83  -1.713249E-81  ...
1.500000E-02  -9.020043E-80  -1.114231E-77  -4.402893E-81  -5.424313E-79  ...
2.000000E-02  -1.201684E-77  -1.467704E-75  -4.549845E-79  -5.533119E-77  ...
2.500000E-02  -1.528953E-75  -1.866265E-73  -4.126064E-77  -5.004886E-75  ...
# ... and so on.
```

## Part 9: Contour-Plot File Format

The contour-plot files are exactly the same as for benchmarks TPV14-15 and TPV18-21.

There are two contour-plot files, one for the main fault and one for the branch fault, as shown here:

Contour-plot files for TPV24 and TPV25	
File Name	Description
cplot_main	Rupture times for the main fault.
cplot_branch	Rupture times for the branch fault

The contour plot file lists the locations of all the nodes on the fault surface, and the time at which each node ruptures.

The contour plot file is an ASCII file that contains three data fields, as follows.

Contour Plot Data Fields for TPV24 and TPV25	
Field Name	Description, Units, and Sign Convention
j	Distance along strike (m). Sign convention: Positive means a location <b>to the right</b> of the junction point. For TPV24 and TPV25, the value of <i>j</i> can range from -16000 to 12000 on the main fault, and from 0 to 12000 on the branch fault. (On the branch fault, <i>j</i> is the distance from the junction point, measured parallel to the strike of the branch fault.)
k	Distance down-dip (m). Sign convention: Zero is the earth's surface, and positive means <b>underground</b> . For TPV24 and TPV25, the value of <i>k</i> can range from 0 to 15000.
t	Rupture time (s). This is the time at which fault slip-rate first changes from 0 to greater than 0.001 m/s. If this node never ruptures, use the value 1.0E+09.

A pair of numbers (*j*, *k*) denotes a point on the fault surface.

The contour plot file consists of three sections, as follows.

<b>Contour Plot File Format for TPV24 and TPV25</b>	
<b>File Section</b>	<b>Description</b>
File Header	<p>A series of lines, each beginning with a # symbol, that gives the following information:</p> <ul style="list-style-type: none"> <li>• Benchmark problem (TPV24 or TPV25)</li> <li>• Author</li> <li>• Date</li> <li>• Code</li> <li>• Code version</li> <li>• Node spacing or element size</li> <li>• Descriptions of data columns (7 lines)</li> <li>• Anything else you think is relevant</li> </ul>
Field List	<p>A <b>single line</b>, which lists the names of the 3 data fields, in column order, separated by spaces. It should be:</p> <p style="text-align: center;">j   k   t</p> <p>(all on one line). The server examines this line to check that your file contains the correct data fields.</p>
Rupture History	<p>A series of lines. Each line contains three numbers, which give the <math>(j, k)</math> coordinates of a node on the fault surface, and the time <math>t</math> at which that node ruptures.</p> <p><b>C/C++ users:</b> We recommend using 14.6E or 14.6e floating-point format.</p> <p><b>Fortran users:</b> We recommend using E15.7 or 1PE15.6 floating-point format.</p> <p>If a node never ruptures, the time should be given as 1.0E+09.</p> <p>Nodes may be listed in any order.</p>

**Note:** The nodes may appear in any order. The nodes do not have to form a rectangular grid, or any other regular pattern.

**Note:** When you upload a file, the server constructs the Delaunay triangulation of your nodes. Then, it uses the Delaunay triangulation to interpolate the rupture times over the entire fault surface. Finally, it uses the interpolated rupture times to draw a series of contour curves at intervals of 0.5 seconds.

Here is an example of a contour-plot file. This is an invented file, not real modeling data.

```
# Example contour-plot file.
#
# This is the file header:
# problem=TPV24
# author=A.Modeler
# date=2013/01/23
# code=MyCode
# code_version=3.7
# element_size=100 m
# Column #1 = horizontal coordinate, distance along strike (m)
# Column #2 = vertical coordinate, distance down-dip (m)
# Column #3 = rupture time (s)
#
# The line below lists the names of the data fields.
# It indicates that the first column contains the horizontal
# coordinate (j), the second column contains the vertical
# coordinate (k), and the third column contains the time (t).
j k t
#
# Here is the rupture history
-6.000000E+02 7.000000E+03 3.100000E-02
-6.000000E+02 7.100000E+03 4.900000E-02
-6.000000E+02 7.200000E+03 6.700000E-02
-7.000000E+02 7.000000E+03 1.230000E-01
-7.000000E+02 7.100000E+03 1.350000E-01
-7.000000E+02 7.200000E+03 1.470000E-01
# ... and so on.
```