CCLDpy (v2.0.0)

Tristan E. Buckreis
tristanbuckreis@ucla.edu
June 19, 2024

Python package for simulating earthquake rupture surface representation for the purpose of computing finite-fault distance metrics (e.g., closest distance to the rupture surface, R_{RUP}; Joyner-Boore distance, R_{JB}), when published finite-fault models from the literature are not available.

The CCLD program was originally coded in Fortran by Brian Chiou and Robert Youngs (Chiou and Youngs 2008; Appendix B), then later updated by Brian Chiou during the NGA-West2 and NGA-Subduction projects (Chiou and Youngs 2008; Contreras et al. 2022). This version of CCLD replicates the original simulation code in Python. Key changes to the program include the following:

1. Added magnitude-scaling relationships for shallow-*crustal* type events (Wells & Coppersmith 1994; Leonard 2014; and Thingbaijam et al. 2017)
2. Added magnitude-scaling relationship for *stable*-continental type events (Leonard 2014). Shallow-crustal type relations for the position of hypocenter along the rupture surface are assumed (Chiou and Youngs 2008).
3. Added magnitude-scaling relationship for *interface* subduction-type events (Thingbaijam et al. 2017). Subduction-type relations for the position of the hypocenter on the rupture surface are assumed (Contreras et al. 2022)
4. Flexibility to specify the number of simulations to use for each magnitude-scaling relation (e.g., use *N* simulations from one scaling-relation or partition *N* simulations into *M* appropriate scaling-relations.

The current version of *ccldpy* (v2.0.0) does not compute distances for real seismic stations, which can be performed using the results of *ccldpy* and the P4CF program (Chiou, B.S-J. 2021).

Magnitude-Scaling Relationships

Scaling-relationships are defined for active shallow-*crustal*, *stable*-continental, *interface*-subduction, and *intraslab* subduction-type earthquakes. These relationships can only be used in their respective tectonic regimes, as summarized herein.

Active Shallow-Crustal Earthquakes

Four scaling-relationships for active shallow-crustal type earthquakes are implemented within *ccldpy*:

1. Wells and Coppersmith (1994): relationships for area (*A*), length (*L*), and width (*W*); one set of coefficients for all styles-of-faulting (i.e., mechanism)
2. Chiou and Youngs (2008): only an aspect ratio (*AR*) relationship; separate coefficients for strike-slip, normal, and reverse styles-of-faulting
3. Lenoard (2014): relationships for *A*, *L*, and *W*; separate coefficients for strike-slip and dip-slip (normal and reverse) styles-of-faulting

4. Thingbaijam et al. (2017): relationships for A, L, and W; separate coefficients for strike-slip, normal, and reverse styles of faulting

Wells and Coppersmith (1994) present separate coefficients for different styles-of-faulting, however the authors do not recommend using these coefficients and instead recommend the "all" set of coefficients. Chiou and Youngs (2008), Leonard (2014), and Thingbaijam et al. (2017) recommend using different sets of coefficients for different styles-of-faulting. Since Chiou and Youngs (2008) only present an AR relationship, it must be used in combination with the other models to develop a rupture surface (details discussed in the *Generating Rupture Geometries* section) of the documentation.

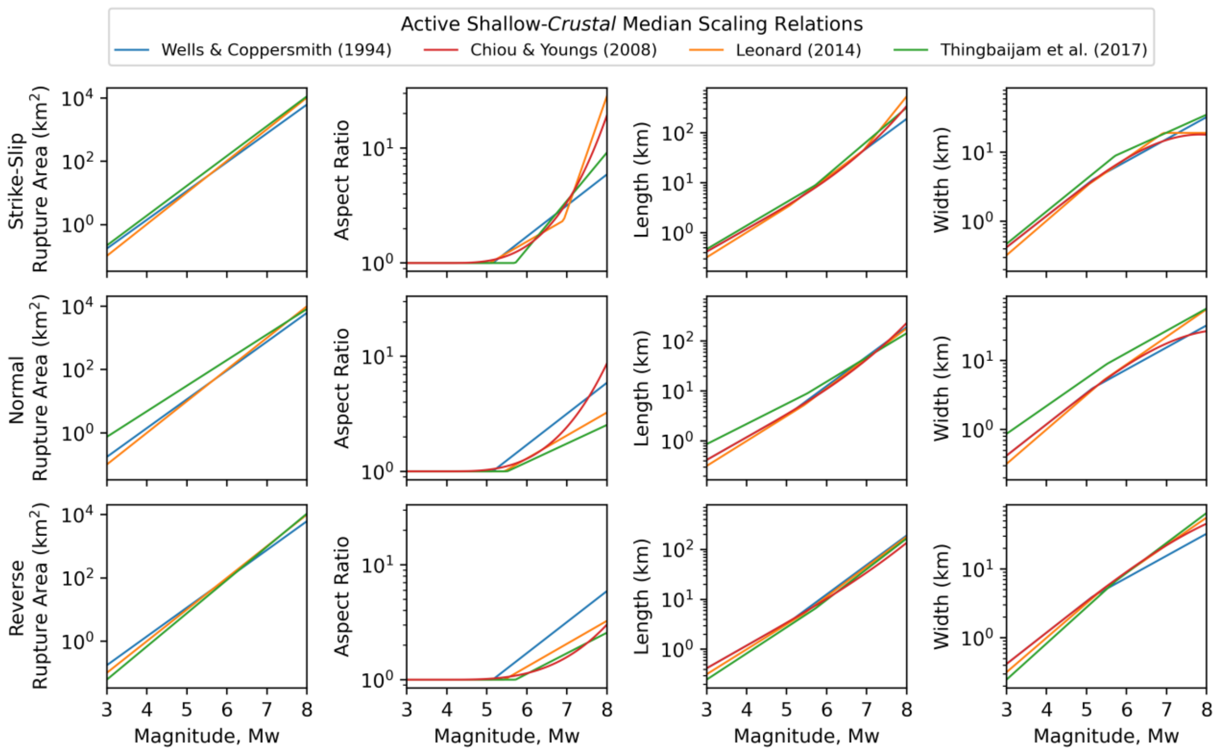

Figure 1: Median magnitude-scaling relationships for active shallow-crustal events.

All of these models were develop using data from events with magnitudes generally greater than 6.0, meaning the relationships must be extrapolated for use with smaller-magnitude events. The L, W, and AR (= L/W) relationships of Wells and Coppersmith (1994), Leonard (2014), and Thingbaijam et al. (2017) for small-magnitudes (generally < 5.5 - 6.0) are not reflective of the as-published relations. This is because extrapolation of these relationships to small-magnitudes leads to unrealistic ruptures (e.g., $AR < 0.1$, which means a fault 1 km long would have a rupture width/depth of 10 km). As a means of addressing these issues, *ccldpy* will compute AR from the as-published L and W relationships. If $AR \leq 1.0$, *ccldpy* will constrain $AR = 1.0$, in effect modeling the rupture as a square which approximates a circular rupture. It follows that L and W can be derived from the published area (A)-relation and the constrained AR-relation as:

$$L = \sqrt{A \times AR}$$

and

$$W = \sqrt{A/AR}$$

Stable-Continental Shallow-Crustal Earthquakes

Only one scaling-relationship for stable-continental shallow-crustal type earthquakes is implemented within *ccldpy*:

1. Leonard (2014): relationships for *A*, *L*, and *W*; separate coefficients for strike-slip, normal, and reverse styles-of-faulting

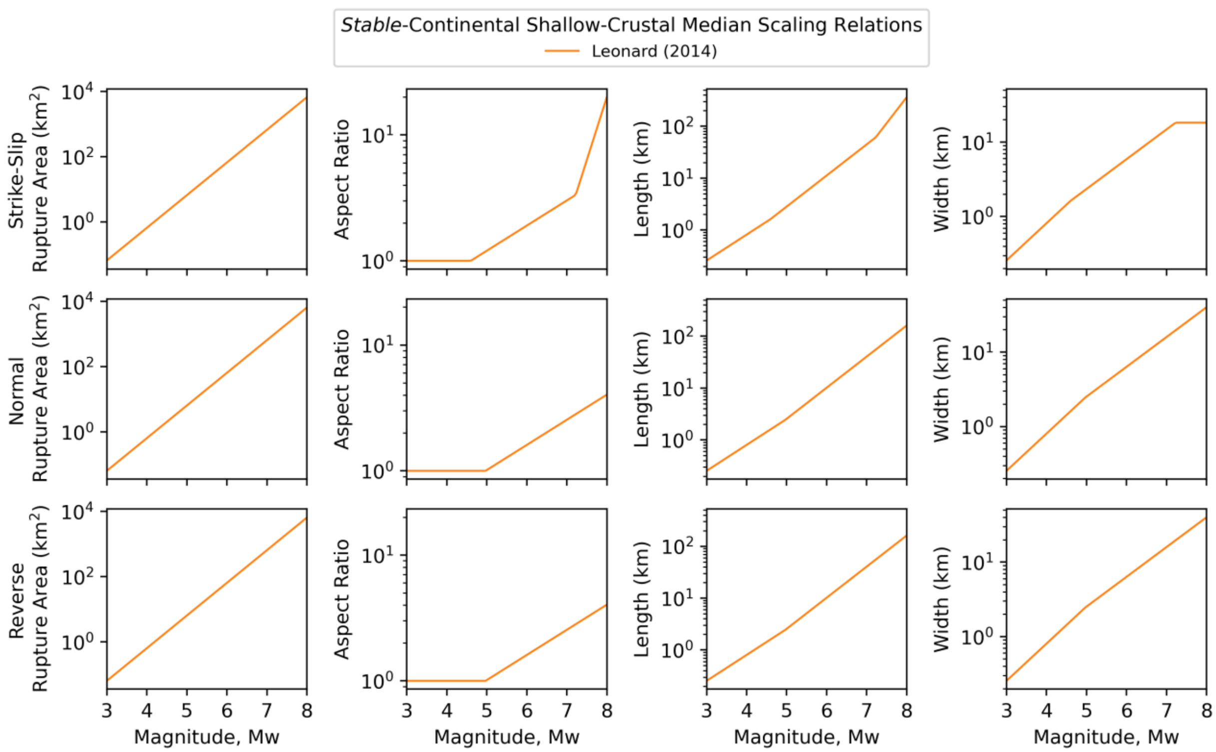

Figure 2: Median magnitude-scaling relationships for stable-continental shallow-crustal events.

Subduction (Interface and Intra-Slab) Earthquakes

Two scaling-relationships are implemented within *ccldy*:

1. Thingbaijam et al. (2017): relationships for *A*, *L*, and *W*; only interface earthquakes
2. Contreras et al. (2022): relationships for *A* and *AR*; separate coefficients for interface and intra-slab type events

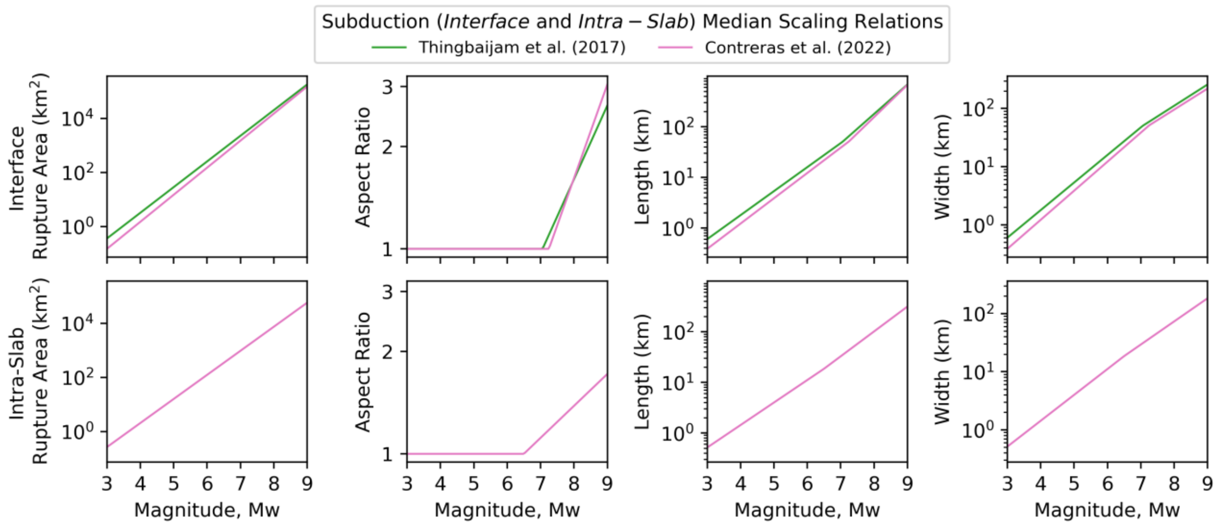

Figure 3: Median magnitude-scaling relationships for subduction (interface and intra-slab) events.

Generating Rupture Geometries

Most scaling relationships provide estimates for rupture area (*A*), length (*L*), and width (*W*), each of which coming from a different model with quantified uncertainty (σ), however Chiou and Youngs (2008) and Contreras et al. (2022) provide relationships for aspect ratio (*AR*) to be used to derive the rupture geometry (*L* and *W*). *ccldy* implements a consistent approach, regardless of which types of information are provided by the published scaling relations. This approach begins by computing *A*, which, for example, can be expressed as:

$$A = 10^{a_1 + b_1 \boldsymbol{M} + \epsilon\sigma_1}$$

where ***M*** is the moment magnitude, a_1 and b_1 are model coefficients, and ϵ is a standard normal variate (mean = 0 and standard deviation = 1). For relationships that provide *AR*, it is simply computed using the published relationship, otherwise *L* and *W* are computed:

$$L = 10^{a_2 + b_2 \boldsymbol{M} + \epsilon\sigma_2}$$

and

$$W = 10^{a_3 + b_3 \boldsymbol{M} + \epsilon\sigma_3}$$

If *L* and *W* were directly computed, *AR* is checked for reasonableness. If *L*/*W* > 1.0, then these dimensions are used for this particular realization of the rupture surface. Otherwise, *AR* is constrained to be 1.0 + $\epsilon\sigma_{AR,CY08}$, where $\sigma_{AR,CY08}$ = 0.16 (Chiou and Youngs 2008). *L* and *W* are then derived from *A* and *AR*.

The position of the hypocenter on the rupture surface is also randomized for each realization. Shallow-crustal and stable-continental type events use relative down-dip (f_d) and along-strike (f_l) distributions which range from 0.0 to 1.0 (inclusive) developed by Chiou and Youngs (2008), whereas subduction-type events (interface and intra-slab) use distributions developed by Contreras et al. (2022). The exact location of the hypocenter (latitude, longitude, and depth) is not randomized in this approach, only the relative position of the hypocenter on the rupture plane is.

Specifying Scaling-Relationships

The scaling-relationships proposed by Wells & Coppersmith (1994), Leonard (2014), Thingbaijam et al. (2017), and Contreras et al. (2022) are self-consistent and provide all the necessary information needed to generate a rupture surface. As such, these models are implemented as separate branches within *ccldpy*. Chiou and Youngs (2008), which is only applicable for shallow-crustal type events, only provides *AR*, which is insufficient by itself to define a rupture geometry. Therefore, the Chiou and Youngs (2008) *AR* relationship can be used with the *A*-relationships proposed by Wells and Coppersmith (1994), Leonard (2014), and Thingbaijam et al. (2017) within *ccldpy* as three separate branches. Table 1 summarizes all current branches implemented in *ccldpy* for each type of earthquake.

Table 1: Summary of scaling-relationship branches currently implemented in *ccldpy*.

Earthquake Type	Model	*A* Relationship	*L* & *W* or *AR* Relationship(s)
crustal	WellsCoppersmith1994	Wells & Coppersmith (1994)	Wells & Coppersmith (1994)
	Leonard2014	Leonard (2014)	Leonard (2014)
	ThingbaijamEtAl2017	Thingbaijam et al. (2017)	Thingbaijam et al. (2017)
	ChiouYoungs2008_WellsCoppersmith1994	Wells & Coppersmith (1994)	Chiou & Youngs (2008)
	ChiouYoungs2008_Leonard2014	Leonard (2014)	Chiou & Youngs (2008)
	ChiouYoungs2008_ThingbaijamEtAl2017	Thingbaijam et al. (2017)	Chiou & Youngs (2008)
stable	Leonard2014	Leonard (2014)	Leonard (2014)
interface	ThingbaijamEtAl2017	Thingbaijam et al. (2017)	Thingbaijam et al. (2017)
	ContrerasEtAl2022	Contreras et al. (2022)	Contreras et al. (2022)
intraslab	ContrerasEtAl2022	Contreras et al. (2022)	Contreras et al. (2022)

Specifying Number of Simulations

The original CCLD program (Chiou and Youngs 2008) performed 101 simulations using a single scaling-relationship for a given earthquake. A major update to *ccldpy* (v2.0.0) is the ability for users to specify the number of simulations, which may include combinations of compatible scaling-relationships (i.e., the same "earthquake type" in Table 1). For example, if a user wants to implement separate branches in the randomization process for a shallow-crustal event using only the Wells & Coppersmith (1994) and Thingbaijam et al. (2017) relationships at 1/3 and 2/3 weights, they can do so by allocating 1/3 of the total simulations to "WellsCoppersmith1994" and 2/3 of the total simulations to "ThingbaijamEtAl2017". It is important to note that the total number of simulations must be odd (in order to extract a true median), so *weights* are entered as "number of simulations". In the example mentioned above, if we assume 101 total simulations, the user can allocate 34 (or 35) simulations to "WellsCoppersmith1994" and 67 (or 66) simulations to "ThingbaijamEtAl2017" which approximate the desired 1/3 and 2/3 weights. If a user specifies an even number of total simulations, *ccldpy* will add an additional simulation to one of the selected relationships for which the user allocated > 0 simulations.

Sensitivity studies suggest that 101 simulations is too few to approximate the expected distributions of rupture geometry (i.e., *A*, *AR*, *L*, and *W*) and stable distributions of $R_{RUP,s}$ for a given site. These studies were conducted by running *ccldpy* for a single set of source input parameters while varying the number of simulations for a single scaling relationship, and were performed for several sets of source input parameters. The distributions/statistics of *A*, *AR*, *L*, *W* and $R_{RUP,s}$ at sample locations were assessed for stability. In other words, the objective was to locate the minimum number of simulations required to achieve the same distributions/statistics from a run which used a large number of simulations. Based on the results of these studies, it is recommended to use at least 300 simulations for each "model" listed in Table 1 in order to ensure the stochastic set of rupture surfaces is representative of the expected distributions.

Selection of the Preferred Rupture Geometry

ccldpy generates a stochastic set of possible rupture surfaces given the available source metadata as outlined above. The objective is to select the most probable surface which does not result in atypically short or long finite-fault distances (e.g., R_{RUP}) for any given site, when published finite-fault models from the literature are not available. This is done by computing rupture distances (R_{RUP}) for a grid of pseudo-stations for each rupture realization (Figure 4). The most-probable rupture surface, which is ultimately selected to compute real distance metrics, is that which minimizes the squared difference between R_{RUP} and the median R_{RUP} at each pseudo-station. In other words, the optimal rupture surface for the purpose of computing reasonable finite-fault distances is that which minimizes the following expression:

$$\sum_{r=1}^{N_r}\sum_{s=1}^{N_s}\left(R_{RUP,median,s} - R_{RUP,r,s}\right)^2$$

where N_r and N_s represent the total number of simulated rupture surfaces and pseudo-stations, respectively; $R_{RUP,r,s}$ is the rupture distance between simulated rupture r and pseudo-station s; and $R_{RUP,median,s}$ is the median rupture distance at pseudo-station s from all simulated rupture surfaces.

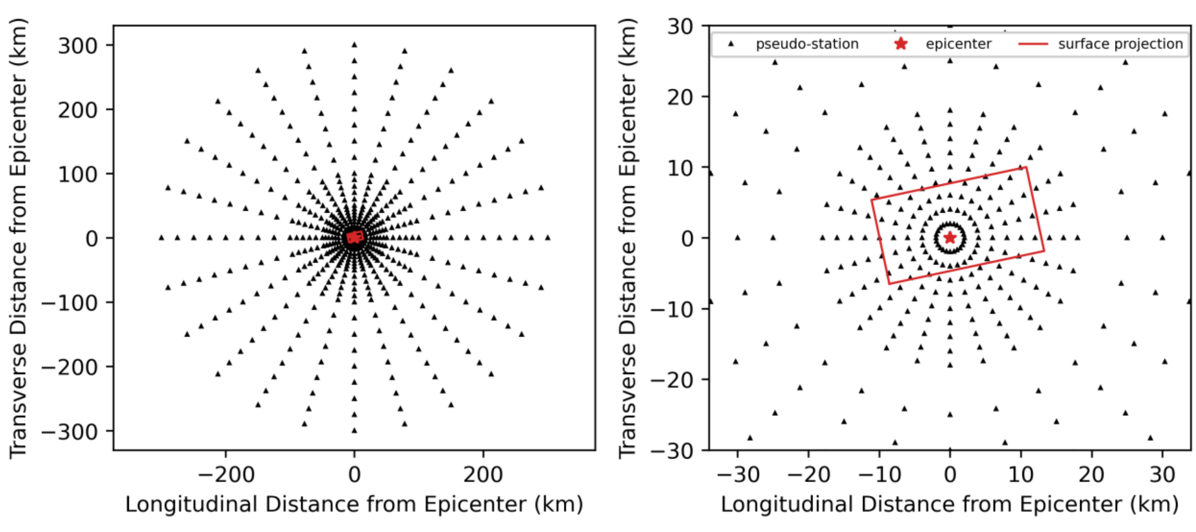

Figure 4: Plan view of grid of pseudo-stations distributed around a realization of a simulated rupture surface.

Installation/Usage

The *ccldpy.py* file can be imported into Python environments.

Simplified instructions

1. Save *ccldpy.py* into your {*desired path*}.
2. Import *ccldpy* into your code:

```
import sys
sys.path.append({desired path})
import ccldpy
```

Simulation Methods

The current version of *ccldpy* (2.0.0) supports five methods of simulation, which are specified using the *method* indicators described below.

A = One or two nodal plane solutions (strike, dip, and rake) are known, however the first solution is preferred. Only the area, aspect-ratio, and position of hypocenter on the rupture surface are randomized between simulations. This method is not recommended because in reality we cannot be certain of a preferred orientation.

B = Two nodal plane solutions are known, however the second solution is preferred. Only the area, aspect-ratio, and position of hypocenter on the rupture surface are randomized between simulations. This method is not recommended because in reality we cannot be certain of a preferred orientation.

C = Two nodal plane solutions are known, and neither is preferred over the other. Each simulation will randomly select which nodal plane solution to use, and the area, aspect-ratio, and position of hypocenter on the rupture surface are also randomized. This is the recommended method when two nodal plane solutions are known.

D = Only one nodal plane solution is known or assumed, but with some uncertainty. The given rake angle is used to assign rupture mechanism (strike-slip, normal dip-slip, or reverse dip-slip - for shallow-crustal or stable-continental type events) and the strike and dip are randomized with $\pm 30^o$ and $\pm 10^o$, respectively. The area, aspect-ratio, and position of hypocenter on the rupture surface are also randomized between simulations. This method is not recommended for general applications, exceptions are when there is outstanding evidence that supports a known or preferred nodal plane solution.

E = No nodal plane solutions are known or assumed. Rake (faulting mechanism), strike [0^o- 360^o), and dip [0^o – 90^o] are randomly assigned with equal probability during each simulation. This is the recommended method when there is missing nodal plane information.

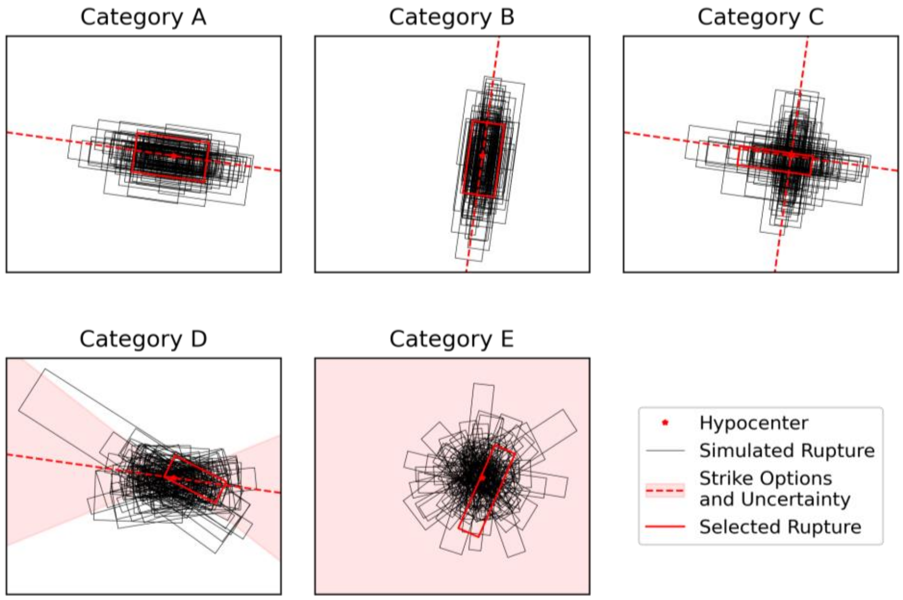

Figure 5: Schematic illustration of simulation results for each simulation *method*.

Main Function

```
simulate_rupture_surface(eqid, eqType, region, elat, elon, hypd,
                         magnitude, method, nsims,
                         mechanism=None,
                         strike=None, dip=None, rake=None,
                         strike2=None, dip2=None, rake2=None)
```

Input Parameters

Required Keys

- `eqid` = unique integer identifier for the event (to set the random seed)
- `eqType` = type of earthquake:
  - "crustal" = shallow-crustal in active tectonic regimes
  - "intraslab" = intraslab-type in subduction regimes
  - "interface" = interface-type in subduction regimes
  - "stable" = shallow-crustal in stable-continental regimes
- `region` = geographic region where the earthquake occurred
  - "japan"
  - "chile"
  - "other"
- `elat` = hypocenter latitude (degrees)
- `elon` = hypocenter longitude (degrees)

- `hypd` = hypocenter depth (km); positive into the ground
- `magnitude` = earthquake moment magnitude (Mw)
- `method` = code for rupture simulation constraints
  - "A" = strike, dip, and rake for first nodal plane solution preferred (optional "strike", "dip", and "rake" arguments are required). Warning: not recommended.
  - "B" = strike, dip, and rake for second nodal plane solution preferred (optional "strike2", "dip2", and "rake2" arguments are required). Warning: not recommended.
  - "C" = strike, dip, and rake are known for two nodal planes, and neither is preferred (optional "strike", "dip", "rake", "strike2", "dip2", and "rake2" arguments are required)
  - "D" = One nodal plane solution for strike, dip, and rake; randomize the strike and dip (optional "strike", "dip", and "rake" arguments are required). Warning: not recommended.
  - "E" = No nodal plane solutions; randomize strike, dip, and rake,(dip and rake are assigned based on faulting mechanism) (if optional "mechanism" argument is not specified, simulations randomly assign one)
- `nsims` = Number of simulations assigned to each M-scaling relationship. Total number of simulations should be odd.
  - `nsims[0]` = Wells & Coppersmith (1994) [recommended 334]
  - `nsims[1]` = Leonard (2014) [recommended 333]
  - `nsims[2]` = Thingbaijam et al. (2017) [recommended 333]
  - `nsims[3]` = Chiou & Youngs (2008) aspect ratio model with Wells & Coppersmith (1994) area relationship [recommended 111 combined with `nsims[3:5]`]
  - `nsims[4]` = Chiou & Youngs (2008) aspect ratio model with Leonard (2014) area relationship [recommended 111 combined with `nsims[3:5]`]
  - `nsims[5]` = Chiou & Youngs (2008) aspect ratio model with Thingbaijam et al. (2017) area relationship [recommended 111 combined with `nsims[3:5]`]
  - `nsims[6]` = Contreras et al. (2022) [recommended 333]

Optional Keys

- `mechanism` = known or preferred style-of-faulting [default None]
  - "SS" = strike-slip: (-180 < rake < -150) or (-30 < rake < 30) or (150 < rake < 180)
  - "NM" = normal: -150 < rake < -30
  - "RV" = reverse: 30 < rake < 150
- `strike` = strike-angle (degrees) of the first nodal plane solution [default None]
- `dip` = dip-angle (degrees) of the first nodal plane solution [default None]
- `rake` = rake-angle (degrees) of the first nodal plane solution [default None]
- `strike2` = strike-angle (degrees) of the second nodal plane solution [default None]
- `dip2` = dip-angle (degrees) of the second nodal plane solution [default None]
- `rake2` = rake-angle (degrees) of the second nodal plane solution [default None]

Returns

- `SIMULATIONS` = pandas DataFrame object containing all simulated rupture surfaces
- `SELECTED` = pandas DataFrame object containing the selected rupture surface and statistics

References

Chiou, B. S.-J. (2021). P4CF [https://github.com/bc88bc/P4CF]

Chiou ,B. S.-J. and Youngs R. R. (2008) *NGA Model for Average Horizontal Component of Peak Ground Motion and Response Spectra*, PEER Rept. 2008/09, Pacific Earthquake Engineering Research Center, Berkeley, California.

Contreras V., Stewart J.P., Kishida T., Darragh R.B., Chiou B. S.-J., Mazzoni S., Kuehn N., Ahdi S.K., Wooddell K., and Youngs R.R., et al. (2020) Source and path database, in *Data Resources for NGA-Subduction Project*, Stewart J. P. (Editor), Chapter 4, PEER Rept. 2020/02, Pacific Earthquake Engineering Research Center, UC Berkeley, Berkeley, California.

Contreras, V., J.P. Stewart, T. Kishida, R.B. Darragh, B.-S.J. Chiou,S. Mazzoni, R.R. Youngs, N.M. Kuehn, S.K. Ahdi, K. Wooddell, R. Boroschek, F. Rojas, and J. Ordenes (2022) NGA-Sub source and path database. *Earthquake Spectra* 38(2): 799 - 840.

Leonard, M. (2014). Self-consistent earthquake fault-scaling relations: Update and extension to stable continental strike-slip faults. *Bulletin of the Seismological Society of America* 104(6): 2953 - 2965.

Thingbaijam K.K.S., Mai P.M., and Goda K. (2017) New empirical earthquake source-scaling laws. *Bulletin of the Seismological Society of America* 107(5): 2225 - 2246.

Veness, C. (n.d.). *Movable type scripts*. Calculate distance and bearing between two Latitude/Longitude

Wells, D.L. and K.J. Coppersmith (1994) New empirical relationships among magnitude, rupture length, rupture width, rupture area, and surface displacement. *Bulletin of the Seismological Society of America* 84(4): 974 - 1002.