

Step 3: Create SRFs for all realisations of the considered faults

In order to run `nhm2srf.py` code to generate SRFs, a "selection file" is needed.

This text file can be in different input formats (depending on the approach utilized for considering uncertainty in CS runs). Here are the three approaches implemented as of Dec 2018:

1. Approach one: The number of realisations for a given fault is prescribed (as done in [Step 2: Determine the number of simulation jobs for a given fault](#)), and the hypocentre location is determined using MonteCarlo randomisation. One slip realisation is generated for a given hypocentre. The format of the input file will be:
`fault_name num_realizations"r"`

note: the "r" key letter is concatenated to the number of realizations. Here is an example for CS18.6: [fault_selection_c18p6_for_SRF.txt](#)

note: We implemented different approaches to create srf realizations (e.g., fixed number of hypocentres, length and width-wise distances between hypocentres). The "r" option referred to "random"/ Monte Carlo approach for place hypocentre, hence the "r" letter tells the code that the hypocentre is placed randomly along the strike and depth

2. Approach two: specify the number of hypocenter along the strike and the number of slip realisations for each hypocentre. The hypocentre is sampled along the strike at the constant depth of $0.6 * (\text{down_dip_width})$. The input file will have this format:
`fault_name num_hypo_along_strike num_slip_realizations_per_hyp`
3. Approach three: specify the distance interval between hypocentres along the strike at the constant depth of $0.6 * (\text{down_dip_width})$. The minimum number of three realisations will be generated. The input file will have this format:
`fault_name hypo_dist_along_strike num_slip_realizations_per_hyp`

See `nhm2srf.py` parameter descriptions for further info on possible input

SRFs can be generated by running `nhm2srf.py` based on the generated "selection file",

go to SrfGen/NHM inside the Pre-processing:

```
> cd /home/hta54/Pre-processing/SrfGen/NHM
```

```
> export PYTHONPATH=$PYTHONPATH:/home/vap30/ucgmsim/Empirical_Engine
```

The code will make use of `/Pre-processing/SrfGen/createSRF.py`, so you need to add it to your python path:

```
> export PYTHONPATH=$PYTHONPATH:/home/hta54/Pre-processing/SrfGen/createSRF.py
```

generate the SRF by the command:

```
> python nhm2srf.py selection_file.txt
```

The above command is sufficient to generate the SRFs for CS. The `nhm2srf.py` can accept other optional arguments. Here is the explanations of those:

Required parameters:

- Fault selection file described above or 'ALL' to include all faults in NHM file.

Optional parameters:

- `-o <dir>` specify output directory
- `--nhm-file <file>` specify alternative NHM fault file
- `--nhm-skip <number>` change number of header lines in NHM file
- `--seed <number>` initial seed
- `--seed-inc <number>` seed increment
- `--nhypo <number>` of hypocentres to generate (equally spaced along strike) if not specified in fault selection file.
- `--nslip <number>` slip distributions per fault to generate if not specified in fault selection file.
- `--dhypo <fraction>` repeated parameter to specify depths of hypocentres eg `--dhypo 0.5 --dhypo 0.75`.
- `-n` or `--nproc <number>` amount of processes to use
- `-p` plot results

