

Progress note on OpenQuake Wrapper with vectorization - new_calc_emp_ds script

This task aims to create a new script to create an emp DS IMDB using the OQ's models that are capable of vectorization to perform better.

A script called `openquake_wrapper_vectorized.py` is designed to work with vectorized OQ models.

The script can handle the following IMs:

- PGA
- pSA

and support the following OQ Models - Green: working fine, Orange - Need some checks

- AG_20
- AG_20_NZ
- ASK_14
- Br_10
- BSSA_14
- CB_14
- CY_14
- K_20
- K_20_NZ
- P_20
- ZA_06

Some models needed some changes to work within a script called `new_calc_emp_ds.py`

Ideally, we were looking for the outputs from the original script, `calc_emp_ds` and a new script, `new_calc_emp_ds` to be identical. (Or reasonably close enough with `np.isclose()`)

1. ASK_14 - Need to check

ASK_14 model requires us to provide a width parameter, and we do not have **width** information.

However, we managed to find an equation that estimates a **width** for us, which is from NGA-West 2 spreadsheet.

```
'Estimations for W and Ztor
If 18 / Sin(dip * Pi / 180) < 10 ^ (-1.75 + 0.45 * M) Then
    West = 18 / Sin(dip * Pi / 180)
Else
    West = 10 ^ (-1.75 + 0.45 * M)
End If
```

This equation is not included in the [ASK_14 paper](#), so we cannot justify proposing these changes to OpenQuake, but we can use this within our OQ wrapper to estimate a **width**.

2. CB_14 - Need to check

CB_14 model also requires us to provide **width**, but unlike ASK_14, the CB_14 [paper](#) includes how to estimate the **width**.

where δ (°) is the dip and W (km) is the width of the fault rupture plane. If W is unknown, it can be estimated from the equation

$$W = \min\left(\sqrt{10^{(M-4.07)/0.98}}, (Z_{BOT} - Z_{TOR}) / \sin(\delta)\right) \quad (39)$$

where Z_{BOT} (km) is the depth to the bottom of the seismogenic crust. The first term in

On top of the **width** estimation, I also implemented an estimation of **Ztor** and **Hypo Depth** in the paper.

given in [Campbell and Bozorgnia \(2013\)](#). If Z_{TOR} is unknown, it can be estimated from Equations 4 and 5 in [Chiou and Youngs \(2014\)](#). If the dimensions of the rupture plane

If the value Z_{HYP} of is unknown, but the geometry and depth of the rupture plane is known or can be assumed, Z_{HYP} can be estimated from the equations

$$\ln \Delta Z = \min(f_{\Delta Z, M} + f_{\Delta Z, \delta}, \ln [0.9(Z_{BOR} - Z_{TOR})]) \quad (35)$$

$$f_{\Delta Z, M} = \begin{cases} -4.317 + 0.984M; & M < 6.75 \\ 2.325; & M \geq 6.75 \end{cases} \quad (36)$$

$$f_{\Delta Z, \delta} = \begin{cases} 0.0445(\delta - 40); & \delta \leq 40 \\ 0; & \delta > 40 \end{cases} \quad (37)$$

which we developed from the metadata in our selected NGA-West2 database ([Campbell and Bozorgnia 2013](#)). In these equations, $\Delta Z = Z_{HYP} - Z_{TOR}$ (km), Z_{BOR} (km) is the depth to the bottom of the rupture plane, and ΔZ (km) is constrained to be no greater than 90% of the

Hence, I opened a PR for **Ztor** estimation on OQ first, but we do not know what will happen. Therefore, the alternative way of using these estimations is to implement these within our wrapper along with ASK_14's width estimation.

3. CY_14

There was a missing piece within this model, and it is now fixed.

Implemented a missing piece in Chiou & Youngs 2014 model

The current Chiou&Youngs 2014 model is missing the following piece which can be found [here](#), page 1144 (28 / 37)

- Predicted PSA value at $T \leq 0.3$ s should be set equal to the value of PGA when it falls below the predicted PGA

We increase the upper bound of the applicable distance range from 200 km to 300 km because of the use of extensive data at distances between 200 km and 300 km (Figure 1). Predicted PSA value at $T \leq 0.3$ s should be set equal to the value of PGA when it falls below the predicted PGA, which could occur at any distance, but is an egregious error mainly at distances larger than 250 km. For application in other active tectonic regions where earthquakes at distances greater than about 50 km are a major contributor to the hazard,

We also found MATLAB code

```
    end
else
    ip_T = find(abs((period- Ti)) < 0.0001);
    [median(i), sigma(i)] = CY_2014_sub(rup.M,ip_
    [PGA, sigma_PGA]=CY_2014_sub(rup.M,2,rup.Rrup
    if median(i)<PGA && period1(i) <= 0.3
        median(i)= PGA;
    end
end
```

4. K_20 and K_20_NZ

Kuehn's model makes some adjustments to the magnitude breakpoint, and this should only happen if the tectonic type is SUBDUCTION_INTERFACE and the region is either Japan or South America. However, the previous version of the K_20 applied this adjustment to every region if the tectonic type is SUBDUCTION_INTERFACE. More details can be found [here](#).

5. ZA_06

ZA_06 officially supports periods between 0.05 and 5.0, but we'd like to compute between 0.01 and 10.0.

Hence, we had to implement both interpolation and extrapolation. Unfortunately, we couldn't have the way OQ's interpolation. (It will include $\log(0)$, which is not valid.). Hence, we keep the interpolation and extrapolation from EE.

Different COEFFS tables

I wrote a script that will compare the DS IMDB between the EE version and the OQ version, and I noticed the model BSSA_14 is the model that is returning utterly different in terms of mean values.

This is because the COEFFS tables were not identical between EE and OQ.

For instance,

```
# EE version
# -1 => PGV, 0 => PGA
periods = np.array([-1, 0, 0.01, 0.02, 0.03, 0.05, 0.075, 0.1])
c3 = np.array([-0.00344, -0.00809, -0.00809, -0.00807, -0.00834, -0.00982, -0.01058, -0.01020])
# OQ version
c3_2 = np.array([-0.003440, -0.008088, -0.008088, -0.008074, -0.008336, -0.009819, -0.010580, -0.010200])

(c3 == c3_2).all() # False
```

Based on the [paper](#) (page 93 or 118/142 in viewer), the **C3** matches EE's C3, not OQ's C3.

Comparison results - between calc_emp_ds and new_calc_emp_ds

The following models do not match (used `np.isclose()`):

- BSSA_14 - As mentioned above, COEFFS table is not quite the same, so this is the expected result.
 - This is because of `_get_basin_depth_term`. Within the OQ, with our usage, Global a.k.a nobasin in OQ term, `basin_depth_term` is basically 0 but within EE, we get something.
 - Unfortunately, paper was not really clear, it did indicate to add `basin_depth_term` for California and Japan, but any other region was not mentioned.
- CB_14 - This is also expected because in the EE version of CB_14 when they estimate width, instead of using either the provided Ztor or estimated one, they estimate Ztor again to calculate width and then set Zhyp(hypo_depth) to 9.0. In contrast, with OQ (once our changes apply), the model will not estimate Ztor again while estimating width, and we do not set Zhyp to 9.0 while calculating width.
- K_20_NZ - ~~This is something different. Previously (calc_emp_ds), we also used the OQ's K_20 model, so nothing to do with EE. The K_20 is fine, but K_20_NZ is different. We need some investigation into this.~~
 - The mystery is solved now, when we generated GNZL, we did not multiply by 1000 on Z1.0.
- ZA_06 - This is expected because of the following behaviour:

```
# In the EE version of ZA_06
# convert to median in g
sa = np.exp(log_sa) / 981
# In the OQ version of ZA_06
from scipy.constants import g
sa = np.exp(log_sa) * 1e-2 / g
# where g = 9.80665
```

```
In [17]: results_df.columns
Out[17]:
Index(['AG_20_NZ_SUBDUCTION_SLAB_ds.db', 'AG_20_SUBDUCTION_SLAB_ds.db',
      'Br_10_ACTIVE_SHALLOW_ds.db', 'BSSA_14_ACTIVE_SHALLOW_ds.db',
      'CB_14_ACTIVE_SHALLOW_ds.db', 'CY_14_ACTIVE_SHALLOW_ds.db',
      'K_20_SUBDUCTION_SLAB_ds.db', 'K_20_NZ_SUBDUCTION_SLAB_ds.db',
      'P_20_SUBDUCTION_SLAB_ds.db', 'ZA_06_SUBDUCTION_SLAB_ds.db'],
      dtype='object')
In [18]: len([False for result in results_df.iloc[:, 3].values if False in result])
Out[18]: 788
In [19]: len([False for result in results_df.iloc[:, 4].values if False in result])
Out[19]: 792
In [20]: len([False for result in results_df.iloc[:, 7].values if False in result])
Out[20]: 792
In [21]: len([False for result in results_df.iloc[:, 9].values if False in result])
Out[21]: 792
```