**New Zealand**

**Probabilistic Seismic Hazard Project**

**User Manual**

Rachel Davidson1, Hasan Manzour1, Natalia Romero2, Linda Nozick2

1University of Delaware

2Cornell University

June 24, 2015

Version 1.1

**Table of Contents**

1. Introduction 1
2. Method 1

2.1. Outputs 1

2.2. Input data 1

2.3. User-specified input parameters 1

2.4. Steps in the method 2

2.4.1. Generate candidate set of probabilistic earthquake scenarios using importance sampling for magnitudes 2

2.4.2. Reduce set of probabilistic earthquake scenarios based on contribution to hazard curves 2

2.4.3. Reduce set of probabilistic earthquake scenarios using optimization-based probabilistic scenario (OPS) method 3

2.4.4. Generate candidate set of probabilistic ground motion maps 4

2.4.5. Reduce set of probabilistic ground motion maps using optimization-based probabilistic scenario (OPS) method 5

1. Software implementation 5

3.1. Overview 5

3.2. Input files 9

3.2.1. Site data (*chch\_pilot\_sites.csv*) 9

3.2.2. Fault source data (*FUN1111.DAT*) 9

3.2.3. Background seismicity source data (*NZBCK211.DAT*) 11

3.2.4. GMPE coefficients from McVerry et al. (2006) (*McVerryT5.csv or McVerryT6.csv*) 12

3.2.5. “True” hazard curve data (*NSHM\_PGA.xyz*) (optional) 12

3.3. User-specified input parameters 13

3.4. Functions 13

3.4.1. *loadingNZ.m* 14

3.4.2.  *QuakeNZ.m*, *QuakeISNZ.m* 14

3.4.3. *SeisEventNZ.m, SeisEvent1NZ.m, SeisEvent3NZ.m* 15

3.4.4.  *PijrNZ.m, PijrNZbin.m* 18

3.4.5. *EQReduceNZ.m* 18

3.4.6. *UpdateMNZ.m*, *UpdateEventNZ.m* 19

3.4.7. *Summary1NZ.m and Summary2NZ.m* 20

3.4.8. *PSHANZ.m* 21

3.4.9.  *GroundMotionNZ.m*  22

3.4.10. *ppdistanceNZ.m* and *pfaultdistanceNZ.m*. 24

3.4.11. *spatialcorre.m* 24

3.4.12. *HazAnaResult1NZ.m and HazAnaResult2NZ.m* 25

3.4.13. *DataGeneration.py* 25

3.4.14. *SolveMe.py* 26

3.4.15. *Print.py* 27

3.5. Output files/Results 28

3.6. Conducting multiple runs 29

4. Matlab/Python commands to implement method 29

4.1. OPS method to generate reduced set of earthquake scenarios only 29

4.2. OPS method to generate reduced set of ground motion maps 30

4.3. Conventional MCS 30

4.4. PSHA 31

5. Potential future extensions 31

References 32

Appendix A. Python and Gurobi software installation notes 33

Appendix B. Soil site classifications 34

Appendix C. Christchurch case study results 35

Appendix D. Gurobi vs COIN comparison 49

**1. Introduction**

This manual describes Appendix C provides results of implementation of the software for an example application for Christchurch.

**2. Method**

The method is briefly summarized here for convenience, and is described in detail in Han and Davidson (2012). It is the method referred to as [1-IS] / [2-OPS] / [3-MinVar] / [4-OPS] in Han and Davidson (2012), but with one modification. Between [1-IS] and [2-OPS], we add a step in which we do a preliminary reduction in the number of candidate earthquake scenarios based on their contribution to the hazard curves (see Section 2.4.2).

**2.1. Outputs**

* A set of earthquake scenarios (defined by rupture source and magnitude), each with an associated hazard-consistent annual occurrence probability, such that together the set represents the complete regional hazard
* A set of ground motion maps, each with an associated hazard-consistent annual occurrence probability, such that together the set represents the complete regional hazard
* Errors between the “true” hazard curves and those based on the reduced sets of earthquake scenarios or ground motion maps
* Errors between the “true” spatial correlation and that based on the reduced set of ground motion maps (only if “true” hazard is computed using convential Monte Carlo simulation; not available if it is given by PSHA)

**2.2. Input data**

* Fault sources
  + Geometry and sense of movement (e.g., reverse, strike-slip), as required by GMPEs
  + Recurrence information (e.g., Mchar, recurrence interval)
* Background seismicity sources
  + Geometry and sense of movement (e.g., reverse, strike-slip), as required by GMPEs
  + Recurrence information (e.g., Gutenberg-Richter *a* and *b* values, Mmin, Mmax)
* Ground motion prediction equation(s) (GMPEs) with any required coefficient values and weights if more than one is used
* Site data
  + Location (lat/long) of interest where hazard curves will be matched and output ground motion maps will be defined
  + Soil information at sites
* “True” hazard curves for each location in the study region (optional; alternatively, can be generated using PSHA or MCS described in Section 3.1)

**2.3. User-specified input parameters**

* Maximum number of earthquake scenarios in output set
* Maximum number of ground motion maps in output set
* Weights to give priority in reducing errors for some return periods or sites
* Return periods at which to match hazard curves
* Ground motion intensity parameter to use (e.g., PGA, Sa(T))

**2.4. Steps in the method**

There are five main steps in the method (Figure 1), described in turn in Sections 2.4.1 to 2.4.5.

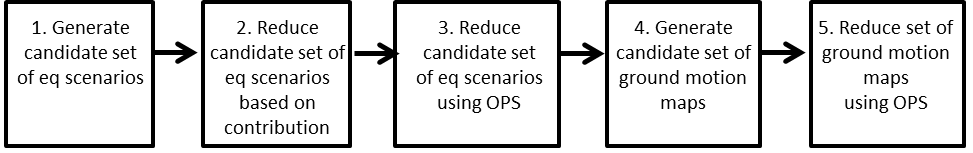


Figure 1. Main steps in method

**2.4.1. Generate candidate set of earthquake scenarios using importance sampling for magnitudes**

Recognizing that conventional Monte Carlo simulation is computationally inefficient because large magnitude events are more important but less frequently sampled than small magnitude events, we use importance sampling to preferentially sample the large magnitude events. Specifically, for each earthquake source, the range in the magnitude density function *dF(M)* is divided into *A* partitions, one earthquake scenario is generated from each partition to get *A* earthquake scenarios, and the importance sampling weight for a magnitude *M* chosen from the *a*th partition (*Ma*, *Ma+1*) provides the hazard-consistent annual occurrence probability for the scenario: . The numerator is the actual probability of a magnitude in *a*th partition; the denominator is the equal importance sampling probability assumed for all partitions. To further force selection of large magnitude events, the partitions may be defined so that they are smaller in higher magnitude ranges, but that has not be found to be necessary. This method is theoretically well-grounded and greatly improves the computational efficiency of conventional MCS.

**2.4.2. Reduce set of probabilistic earthquake scenarios based on contribution to hazard curves**

In this Step, we do an initial reduction of the candidate set of earthquake scenarios by computing the average contribution of each earthquake scenario to the hazard curves (Eq. 1), and selecting only those with the highest contributions, say those that together account for say 90% of the contribution. The contribution *Cj* of earthquake scenario *j* to matching the “true” hazard across the region is defined as in Han and Davidson (2012):

(1)

where is annual occurrence probability of earthquake scenario *j*; is the “true” ground motion for return period *r* at site *i*; is the ground motion value at site *i* caused by earthquake *j*; and *I*, *J*, and *R* are the number of sites, earthquake scenarios, and return periods, respectively. The term in brackets is the percentage contribution of earthquake scenario *j* to the hazard at site *i* and return period *r*, and *Cj* is the average contribution of scenario *j* over all sites and return periods. In Han and Davidson 2012, the selected earthquake scenarios were then renormalized and this step was used instead of the OPS method described in Section 2.4.3. Here we use it before and in addition to the OPS method. Since the OPS method recomputes the annual occurrence probabilities, there is no need to do that in this step. In this step, we only choose the earthquake scenarios in the first reduced set.

While we use the OPS method in Step 3 to optimally reduce the set of earthquake scenarios (Section 2.4.3), the optimization can be solved more easily if the candidate set is not huge, as it may be in a region of very high seismicity (like New Zealand), even after using importance sampling of the magnitudes. Thus, this second step has been added since Han and Davidson (2012). Since a relatively small percentage of the earthquake scenarios typically account for the vast majority of the contribution to the hazard curves, there is virtually no negative effect to removing most of the earthquake scenarios from the candidate set in this step, and the benefit is easier computation in the optimization in Step 3. Further, this step is straightforward and quick to apply since it only requires computing Equation 1 for every earthquake scenario *j*, sorting them in descending order from highest to lowest , and then choosing the ones that account for the highest say 90% or 95%.

**2.4.3. Reduce set of probabilistic earthquake scenarios using optimization-based probabilistic scenario (OPS) method**

In the optimization-based probabilistic scenario (OPS) method, a mixed-integer linear optimization model selects a reduced set of earthquake scenarios from the candidate set developed in Step 1, and determines hazard-consistent annual occurrence probabilities for each earthquake scenario in the reduced set so as to make the regional hazard estimated by the earthquake scenarios in the reduced set match the “true” regional hazard as well as possible (Figure 2).



Figure 2. Schematic defining errors between the “true” and reduced set hazard curves for site *i*

The objective of the optimization is to minimize the sum of the errors over all sites *i* and return periods *r*, between points on the “true” hazard curves and the corresponding points on hazard curves developed with the reduced set of earthquakes and hazard-consistent annual occurrence probabilities:

(2)

where and are the errors resulting from overestimating and underestimating, respectively, the “true” hazard curve for return period *r* at site *i* (Figure 2); and *w­ir*≥0 is the user-specified weight given to the error associated with site *i* return period *r*. The errors are positive if the “true” values are overestimated and zero otherwise; are positive if the “true” values are underestimated and zero otherwise. The weights *wir* may be included to force errors to be smaller for certain sites *i* or return periods *r* of particular interest. Due to the nonlinear shape of a typical hazard curve, probability errors associated with higher exceedence probabilities (i.e., lower return periods) will be larger than those associated with lower exceedence probabilities. In the empirical analysis in this paper, we let all *wir*=*r* so that we give equal weight to each error as a percentage of its associated true exceedence probability (1/*r*), i.e., =.

Suppose *Pj* is the hazard-consistent annual occurrence probability for earthquake scenario , *J* is the number of candidate earthquakes, *Yir* is the ground shaking from the “true” hazard curve for return period *r* at site *i*, and *yij* is the ground shaking at site *i* caused by earthquake *j*. The input coefficients can be calculated by integrating the ground motion prediction equation. The following constraint defines the error terms, for each site-return period combination, as the difference between the “true” annual exceedence probability, 1/*r*, and the annual exceedence probability estimated using the reduced set of earthquakes:

(3)

Let be a binary variable that indicates whether or not earthquake scenario *j* is included in the selected reduced set of events. The next two constraints then allow the user to limit the number of earthquakes that are allowed to be included in the reduced set to a user-specified number *Jred*:

(4)

(5)

Finally, the decision variables are constrained as follows:

(6)

(7)

(8)

If , earthquake *j* is not included in the reduced set of earthquake scenarios. The model also provides the hazard curve errors, and , for each site *i* and return period *r*, so the user can see how big the errors are and how they are distributed.

**2.4.4. Generate candidate set of probabilistic ground motion maps**

In conventional MCS, we sample the same number of ground motion maps for each earthquake scenario. To be more efficient, however, we can vary the number of maps *Nj* sampled for each earthquake scenario *j* so as to minimize the variance in the estimated ground motion exceedence probabilities. Intuitively, for earthquake scenarios with very large magnitudes close to the study area (or small magnitudes far from the study area), the probability of exceeding a specified ground motion does not vary much across ground motion maps, so there is no need to sample a lot of them.

Our goal is to determine *Nj*, the number of maps to sample from each earthquake scenario *j* so as to minimize the variability in the estimated probability of exceeding ground motion *Yir*, while ensuring that a user-specified *N* ground motion maps are sampled in total (). Following the derivation in Han and Davidson (2012), we determine *Nj* should be as defined in Equation 9, and the probability of each ground motion map is .

(9)

This method minimizes the variance for only one specified site *i* and return period *r*. The results may be different for each site *i* and return period *r*. To get the final set of *Nj* values, therefore, we average the *Nj* values over all sites *i* and return periods *r*. If a resulting value of *Nj* is non-integer, we round to the nearest nonzero integer.

Note: Matlab functions are included either to generate different numbers of ground motion maps for each earthquake scenario as described in this section, or to generate the same number for each.

**2.4.5. Reduce set of probabilistic ground motion maps using optimization-based probabilistic scenario (OPS) method**

In this step we reduce the size of the set of ground motion maps using the same optimization formulation presented in Step 3. There are only two differences. First, in this application, we replace earthquake scenario *j* with ground motion map *n*. Second, in this application the values of are binary (0 or 1) values since there is no uncertainty in the ground motion at site *i* in ground motion map *n*. Although the formulation is the same, in this application, the problem is often more difficult to solve. As a result, it is easier to get a good solution if the candidate set of ground motion maps is as small as possible, and thus it is beneficial to be efficient in Steps 1 to 4.

**3. Software implementation**

**3.1. Overview**

The method is implemented for New Zealand using Matlab R2013b, Python 2.7.6, and optionally, Gurobi 6.0. Python is used only to create and call the optimization model in both Steps 2.4.3 and 2.4.5. The optimization is solved using either Gurobi or the PULP\_CBC\_CMD solver from COIN-OR. The Gurobi Optimizer is a state-of-the-art solver for mathematical programming that can be called via many programming languages, including Python, C++, and Matlab. (We used Python because of its ease-of-use for this purpose). However, while Gurobi is free for academic use, it is not free for all users. Thus, we also used the PULP\_CBC\_CMD solver, a free open source solver available from COIN-OR (referred to as the COIN solver in this document). COIN-OR, which stands for COmputational INfrastructure for Operations Research, is an open source initiative (www.coin-or.org). Once the optimization model is created in Python, choosing which solver to use is simply a matter of naming the choice in the *SolveMe.py* function (Section 3.4.14). Matlab is used for all other computations, including for example, reading input, sampling earthquake scenarios and ground motions, computing hazard curves, and generating output. Appendix A includes information on installing the Python and Gurobi software, including necessary libraries. Appendix D includes some comparison of the two solvers’ performance.

The model implementation presented here is based on an implementation developed by Yeliang Han and Rachel Davidson for Los Angeles, California and described in Han and Davidson (2012) and *Seismic Hazard User Mannual\_Yeliang.docx*. Some of the names of files and functions are a result of the legacy of the earlier implementation (e.g., NZ is included at the end of all .m files).

The input hazard information is from the New Zealand National Seismic Hazard Model. It uses the McVerry et al. (2006) ground motion prediction equations and site effects, distributed seismicity sources are treated as point sources with recurrence relations that are not time-dependent. There is an option to assume the fault source recurrence relationships are delta functions or truncated normal distributions. There is also an option to include the spatial correlation in the ground motion residuals using the Jayaram and Baker (2009) model, or to assume no correlation.

Figure 3a and 3b summarize the steps required to implement the method, including the functions to call in red, and for each step, the main input and output variables. The step numbers refer to the five steps described in Section 2.4. In Step 0, the input data are loaded, including information on sites, fault sources, background sources, and if desired, the “true” hazard curves. The “true” hazard curve information may come from a separate PSHA or conventional Monte Carlo Simulation (MCS). Functions are provided to generate the “true” hazard curves using conventional MCS (see steps in Figure 43) or PSHA (see steps in Figure 5), but it is not necessary to do either of those if “true” hazard curves are already provided. An advantage of conducting a conventional MCS to get the “true” hazard is that it allows evaluation of the spatial correlation errors as well as hazard curve errors.

In Step 1, the candidate set of earthquake scenarios is simulated, using importance sampling of the magnitudes. This produces a matrix *M* that describes the set of candidate earthquake scenarios. Step 2, reducing the set of candidate earthquake scenarios based on hazard curve contributions, requires running a few functions. In Step 2a, the mean and inter- and intra-event residuals are computed for each candidate earthquake scenario by setting *num* equal to the number of earthquake scenarios in *M*. Note that *SeisEvent1NZ.m* generates one or more specific ground motion maps for each earthquake scenario, but in this step, we only care about the mean and residuals, which will be used to compute the exceedance probability in Step 2b. In Step 2b, we integrate over the ground motion distribution for site *i* and earthquake scenario *j* to calculate , the probability the ground motion at each site *i* in earthquake scenario *j* exceeds the “true” ground motion at site *i* with return period *r*, for all combinations of *i*, *j*, *r*. These probabilities are a main input to the reduction based on hazard curve contributions (Step 2c) and to the optimization model, which is solved in Step 3, respectively). The first reduction in the set of earthquake scenarios is in Step 2c, in which the set of earthquake scenarios necessary to reach a user-specified cumulative percentage of the hazard curve contribution are identified.

In Step 3, the optimization model is solved to provide the final reduced set of earthquake scenarios and associated adjusted annual occurrence probabilities, which are stored in *RedM2*. Step 3a is conducted in Python. *DataGeneration.py* first reads the input data, then *SolveMe.py* creates the optimization model and calls either Gurobi or COIN to solve it, and finally *Print.py* generates the output files. Step 3b simply uses the *Pj* values output by the optimization, *Pjs*, (assigned annual probability of each earthquake scenario *j*) to update the reduced set of earthquake scenarios and stores them in *RedM2*.

In Step 4, a set of candidate ground motion maps is generated, with the number of maps per earthquake scenario determined so as to minimize sampling variance and result in the specified total number of ground motion maps for the whole set.

The optimization is run again in Step 5, this time to select the final set of ground motion maps. This requires running three functions. Step 5a simply generates the binary P matrix required as input for the optimization. Step 5b is the second of two steps in which Python is used. As with Step 3a, *DataGeneration.py* first reads the input data, then *SolveMe.py* creates the optimization model and calls either Gurobi or COIN to solve it, and finally *Print.py* generates the output files. Similar to Step 3b, Step 5c simply uses the *Pj* values output by the optimization, *Pjs2*, (assigned annual probability of each earthquake scenario *j*) to update the reduced set of ground motion maps and stores them in *RedEvent*.

The functions *Summary1NZ.m* and *Summary2NZ.m* can be used to generate the hazard curve (and if applicable, spatial correlation) errors and results to map. The former generates errors for a set of earthquake scenarios; the latter for a set of ground motion maps.



Figure 3a. First steps required to implement method, including functions to run (red), and main input and output variables from each (blue). Step numbers refer to four steps in Section 2.4

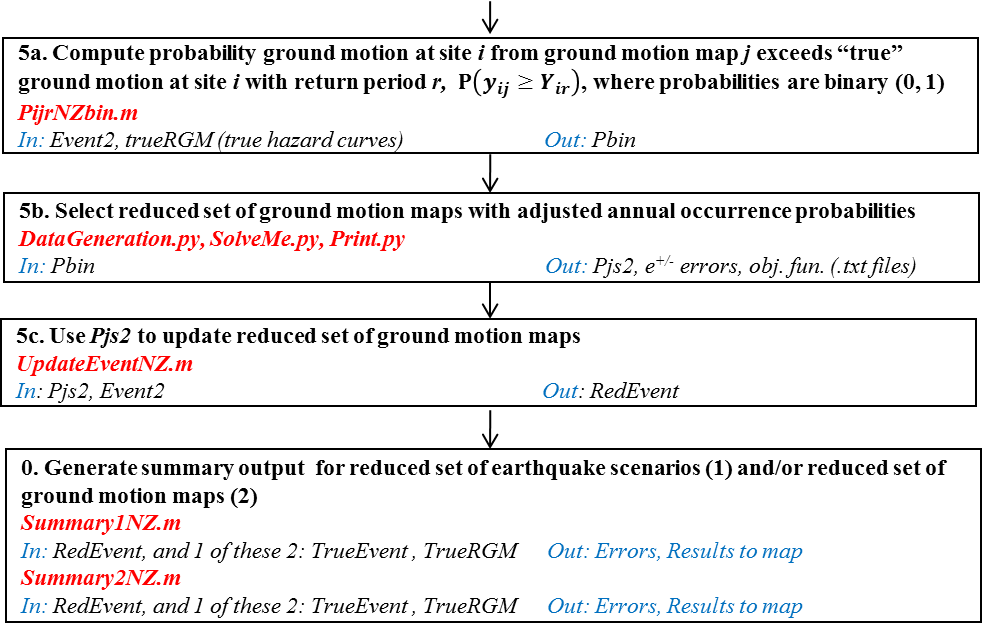


Figure 3b. Last steps required to implement method, including functions to run (red), and main input and output variables from each (blue). Step numbers refer to four steps in Section 2.4

Figure 4 summarizes the steps required to generate a “true” set of ground motion maps using conventional Monte Carlo simulation. This can be used if an alternative set is not available and/or if the user wants to assess the spatial correlation errors in the reduced set. After loading the data in Step 0, a set of earthquake scenarios are simulated in Step 1 for a user-specified number of years. In Step 2, one ground motion map is simulated for each earthquake scenario.

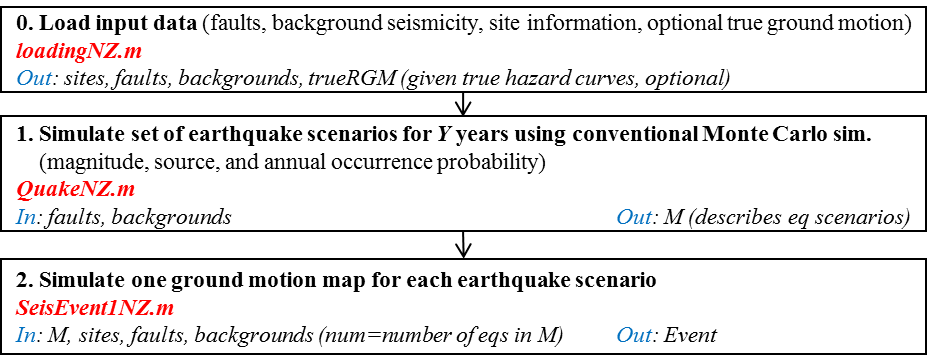


Figure 4. Steps required to generate “true” hazard using conventional Monte Carlo simulation

Figure 5 summarizes the steps required to generate a “true” set of ground motion maps using PSHA. This can be used if an alternative set is not available. After loading the data in Step 0, the hazard curve is generated for each site in Step 1 by integrating over the ground motion distribution.

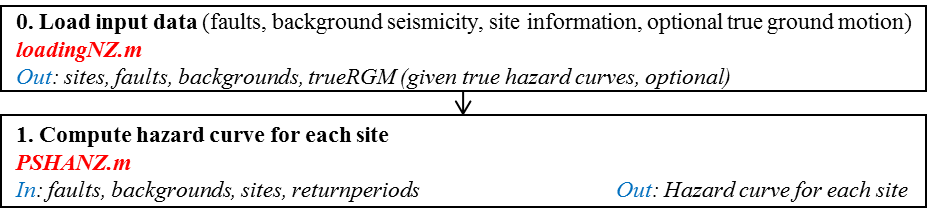


Figure 5. Steps required to generate “true” hazard using PSHA

**3.2. Input files**

The following five input files are required, each of which is described in turn. They were used in the same format sent by GNS.

* Site data (*chch\_pilot\_sites.csv*)
* Fault source data (*FUN1111.DAT*)
* Background seismicity source data (*NZBCK211.DAT*)
* GMPE coefficients (*McVerryT5.csv* or *McVerryT6.csv*)
* “True” hazard curve information (*NSHM\_PGA.xyz*) (optional)
  + 1. **Site data (*chch\_pilot\_sites.csv*)**

The site data file (*chch\_pilot\_sites.csv*) contains information about the sites at which the hazard will be matched and at which the final ground motion maps will be defined (Table 1). It includes a row for each site of interest and the following three **columns**:

1. *lon*. Longitude of the site in degrees
2. *lat*. Latitude of the site in degrees
3. *siteclass*. Category defining the site class (A, B, C, D, or E) as defined in McVerry et al. (2006), Table 3 (included in this document as Appendix B)

Table 1. Example excerpt from a site data file



* + 1. **Fault source data (*FUN1111.DAT*)**

The fault source data file (*FUN1111.DAT*) contains geometry, sense of motion, and recurrence information for each fault source. It includes 3 header rows of data which was not used, then a block of rows for each fault source (Table 2). The number of rows in the fault blocks vary depending on how many segments each includes. Each fault block includes the following **rows** of data:

1. *fault name*, *sense of movement* (if=interface subduction, nn=normal, ns=normal-strike slip, nv=normal volcanic, rv=reverse, rs=reverse-strike slip, ss=pure strike slip, sn=strike slip normal, sr=strike slip reverse)
2. *number of segments* (followed by “D” that is not used)
3. *dip* (in degrees), *dip direction* (in degrees), *depth to bottom of fault* (km), *depth to top of fault* (km). Dip direction is defined as azimuth fault is dipping towards (e.g., 270 means the fault dips to the west)
4. *start point latitude* (44 4.2), *start point* *longitude* 169 42.0, *end point latitude* (44 26.7 ) *end point longitude* (169 41.5) (all in degrees and decimal minutes), *characteristic magnitude* (Mw), *recurrence interval* (years). Note this first line is the start and end points of the multi-segment fault so the start point is the same as the start point of segment 1, and the end point is the same as the end point of segment n.
5. *segment 1* *start point latitude*, *segment 1* *start point longitude*, *segment 1* *end point latitude*, *segment 1* *end point longitude* (all in degrees and decimal minutes)
6. *segment 2* *start point latitude*, *segment 2* *start point longitude*, *segment 2* *end point latitude*, *segment 2* *end point longitude* (all in degrees and decimal minutes)

…

4+n. *segment n* *start point latitude*, *segment n* *start point longitude*, *segment n* *end point latitude*, *segment n* *end point longitude* (all in degrees and decimal minutes)

5+n. -1 = end of fault code block

Note that the fault trace coordinates’ order may NOT follow the “right-hand rule” which states that if looking from the start to the end of the fault, the fault dips to the right. Instead, the program uses the dip direction to determine which azimuth the fault is dipping.

Table 2. Example excerpt from top of a fault source data file (first three rows are not used)



* + 1. **Background seismicity source data (*NZBCK211.DAT*)**

The background seismicity source data file (*NZBCK211.DAT*) contains geometry, sense of motion, and recurrence information for each background seismicity source. It includes one header row of data, then a row for each background seismicity source (Table 3).

The first row contains the following quantities. Note that t1, t2, t3 are the same as in Stirling, except the present year is 2010 instead of 1997.

1. *magnitude of completeness 1*, Mc1 (known as *magmin1* as in Stirling 2002, p1886, Eq. 6)
2. *time of completeness 1*, t1, present year-1964 (known as *ctime1* in Stirling 2002, p1886, Eq. 6)
3. *magnitude of completeness 2*, Mc2 (known as *magmin2* as in Stirling 2002, p1886, Eq. 6)
4. *time of completeness 2*, t2, 1964-1940 (known as *ctime2* in Stirling 2002, p1886, Eq. 6)
5. *magnitude of completeness 3*, Mc3 (known as *magmin3* as in Stirling 2002, p1886, Eq. 6)
6. *time of completeness 3*, t3, 1940-1840 (known as *ctime3* in Stirling 2002, p1886, Eq. 6)
7. *seismicity grid spacing* (decimal degrees)

After the first row, there is a row for each background seismicity source and the following 12 **columns**:

1. *number of events > Mc1 for period 2010-t1* (i.e., 1964-2010) (known as *N1* as in Stirling 2002, p1886, Eq. 6)
2. *number of events > Mc2 for period 2010-t1-t2 to 2010-t1* (i.e., 1940-1964) (known as *N2* as in Stirling 2002, p1886, Eq. 6)
3. *number of events > Mc3 for period 2010-t1-t2-t3 to 2010-t1-t2* (i.e., 1840-1940) (known as *N3* as in Stirling 2002, p1886, Eq. 6)
4. *Gutenberg-Richter b value*
5. *Maximum magnitude*, Mmax (in Mw)
6. *placeholder*
7. fault type
8. *placeholder*
9. *placeholder*
10. *latitude* (decimal degrees south)
11. *longitude* (decimal degrees)
12. *depth* (km)

Table 3. Example excerpt from top of a background seismicity source data file



The header row together with the first 3 columns of data for each background source are used to compute the Gutenberg-Richter *a* value using Equation 6 from Stirling et al. (2002, p1886), repeated here for convenience:

(10)

where: (11a)

(11b)

(11c)

Note that the program assumes *Mmin*=5.25 for all background sources (set in line 205 of *loadingNZ.m*, Section 3.4.1).

* + 1. **GMPE coefficients from McVerry et al. (2006) (*McVerryT5.csv* or *McVerryT6.csv*)**

The GMPE coefficient file contains the coefficients required for the McVerry et al. (2006)ground motion prediction equations. There are two possible choices—*McVerryT5.csv* or *McVerryT6.csv*, corresponding to the coefficients in Table 5 or 6 from McVerry et al. (2006), respectively. The coefficients in Table 5 should be used to compute the stronger horizontal component of ground motion intensity. The coefficients in Table 6 should be used to compute the geometric mean of the ground motion intensity components. Table 6 likely will be used in general, but Table 5 was used to compare to the National Hazard Maps for the Christchurch case. Both tables have the same format. In each case, the file includes a row for each possible ground motion intensity metric (i.e., PGA, PGA’, Sa(0.075), Sa(0.1), Sa(0.2), Sa(0.3), Sa(0.4), Sa(0.5), Sa(0.75), Sa(1.0), Sa(1.5), Sa(2.0), Sa(3.0), where periods are in seconds). There are 26 columns. The first 22 contain coefficients C1, C2, etc. The next four contain SigmaM6, Sigslope, Tau, and SigtotM6.

* + 1. **“True” hazard curve data (*NSHM\_PGA.xyz*) (optional)**

The “true” hazard curve data file contains what will be considered the “true” hazard curves that will be matched in Steps 3 and 5 of the method. It is optional. If it is supplied, it may come from a PSHA or conventional MCS analysis done separately. If it is not supplied, the user can use the functions provided to develop a MCS or a PSHA assessment of the “true” hazard (Figures 4 and 5, respectively). A benefit of using the functions provided to generate your own MCS assessment of the “true” hazard rather than using a user-provided *NSHM\_PGA.xyz* file is that it allows computation of spatial correlation errors. If user-provided “true” hazard curves are used, it is not possible to compute spatial correlation errors.

This file contains a row for each site that will be matched, and the following 6 **columns** of data:

1. longitude (degrees)
2. latitude (degrees)
3. PGA (g) for 250-year return period for site class C
4. PGA (g) for 500-year return period for site class C
5. PGA (g) for 1000-year return period for site class C
6. PGA (g) for 2500-year return period for site class C

Table 4. Example excerpt from a *NSHM\_PGA.xyz* “true” hazard curve data file



* 1. **User-specified input parameters**

Table 5. User-specified input parameters

|  |  |
| --- | --- |
| **Variable** | **Name** |
| *maxD* | Maximum distance from study area at which to consider sources (km) |
| *inputfolder* | Location of folder with all input files and .m function files |
| *distMchar* | Indicates which recurrence relation distribution to use in sampling earthquake scenarios for fault sources (0=delta; 1=truncated normal) |
| *Tsec* | Period (seconds) of specified ground motion intensity parameter (*Tsec*=0 refers to peak ground acceleration; otherwise *Tsec* refers to spectral acceleration in g at period *Tsec* seconds, *Sa(T)*) |
| *CorrCoef* | Indicates if spatial correlation among residuals should be included as modeled in Jayaram and Baker (2009) (1) or not (0) |
| *num* | Desired total number of simulated ground motion maps across all earthquake scenarios (in *SeisEvent1NZ.m* and *SeisEvent3NZ.m*) |
| *Jred* | Desired number of earthquake scenarios (ground motion maps) included in reduced set of earthquake scenarios (ground motion maps) (in *SolveMe.py*) |
| *returnperiods* | Vector of return periods at which to match “true” hazard curves, in years |

**3.4. Functions**

The following functions are required to implement this method and are described in turn in this section. Each subsection includes a description of what the function does, the input files and/or parameters it requires, the output it generates, and any additional notes on how it works.

Matlab functions you call directly (except *SeisEventNZ.m*)

* + - 1. *loadingNZ.m*
      2. *QuakeNZ.m, QuakeISNZ.m*
      3. *SeisEventNZ.m, SeisEvent1NZ.m, SeisEvent3NZ.m*
      4. *PijrNZ.m, PijrNZbin.m*
      5. *EQReduceNZ.m*
      6. *UpdateMNZ.m, UpdateEventNZ.m*
      7. *Summary1NZ.m, Summary2NZ.m*
      8. *PSHANZ.m*

Matlab functions called by other functions

* + - 1. *GroundMotionNZ.m*
      2. *ppdistanceNZ.m* and *pfaultdistanceNZ.m*
      3. *spatialcorre.m*
      4. *HazAnaResult1NZ.m*, *HazAnaResult2NZ.m*

Python functions

* + - 1. *DataGeneration.py*
      2. *SolveMe.py*
      3. *Print.py*

**3.4.1. *loadingNZ.m***

This function imports the input files described in Section 3.2 and stores the data in variables for later use. To save space, fault sources, background seismicity sources, and “true” hazard data (if included) are only saved if they are within the user-specified distance *maxD* of the study area boundaries. This procedure is designed assuming a rectangular study area defined by the site data. The function calls the functions *ppdistanceNZ.m* and *pfaultdistanceNZ.m* to compute the point-to-point distances and point-to-fault distances necessary to determine if background and fault sources, respectively, are within the specified distance *maxD*. It assumes the entire fault ruptures for fault sources, and background seismicity sources are treated as point sources. For efficiency, the *Rrup* distances are saved for each fault and background source for later use[[1]](#footnote-1). It also assumes *Mmin*=5.25 for all background sources (set in line 213 of *loadingNZ.m*).

Call:[ faults,backgrounds,sites,trueRGM ] = loadingNZ( inputfolder, maxD );

Input parameters:

* + - * *inputfolder*. Location of folder with input files and .m files
      * *maxD*. Maximum distance from study area at which to consider sources (km)

Input files: *chch\_pilot\_sites.csv*, *FUN1111.DAT*, *NZBCK211.DAT*, *McVerryT6.csv*, *NSHM\_PGA.xyz* (optional)

Output variables:

* *sites*. Structure with row for each site *i*, and the following fields: latitude, longitude, and soil type.
* *faults*. Structure with row for each fault within *maxD* of study area, and the following fields: name, sense of movement, number of segments, dip (deg), dip direction (deg), depth of bottom of fault (km), depth of top of fault (km), coordinates defining start and end of fault, characteristic magnitude, return period, coordinates defining all segments (lat/long), 1x*(numsites)* vector containing the *Rrup* (km) from each site to the fault
* *backgrounds*. Structure with row for each background seismicity source within *maxD* of study area, and the following fields: Gutenberg-Richter *b* value, *Mmin*, *Mmax*, sense of movement, latitude, longitude, depth (km), Gutenberg-Richter *a* value (computed as described in Section 3.2.3), 1x*(numsites)* vector containing the *Rrup* (km) from each site to the source
* *trueRGM*. For each site *i*, ground motion intensity (g) for 250-, 500-, 1000-, and 5000-year return periods (optional)

**3.4.2. *QuakeNZ.m****,* ***QuakeISNZ.m***

*QuakeNZ.m* and *QuakeISNZ.m* are alternative functions for simulating a set of earthquake scenarios (defined by source and magnitude), each with an associated probability of occurrence. In both functions, for fault sources, if *distMchar*=0, it is assumed that the each fault source only generates earthquakes of magnitude *Mchar* and thus only one scenario is sampled from each. If *distMchar*=1, it is assumed that the magnitude distribution is a truncated normal distribution with *µ*=*Mchar*, *σ*=0.06, and range from (*Mchar* - 0.2) to (*Mchar* + 0.2). In both functions, for background sources, the following recurrence relation is used: (and ), where *a* is as in Eq. 10.

***QuakeNZ.m***uses conventional Monte Carlo simulation (MCS) to simulate a set of earthquake scenarios for a specified number of years. It can be used to generate a set of “true” ground motion maps if one is not available and/or if one wants to examine spatial correlation errors (Figure 4).

Call: M=QuakeNZ(years, faults, backgrounds, distMchar)

Input parameters:

* *years*. Number of years to run simulation
* *distMchar*. Indicates which recurrence relation distribution to use in sampling earthquake scenarios for fault sources (0=delta; 1=truncated normal)

Input variables: *faults*, *backgrounds* (both output by *loadingNZ.m*)

Output variables: *M.* Matrix including a row for each earthquake scenario generated, and the following 5 columns of information:

1. Year of earthquake scenario
2. ID of the background source or fault that generated the earthquake scenario
3. Magnitude of earthquake scenario
4. Indicator of whether it was generated by a background source (0) or a fault source (1)
5. Annual occurrence probability

***QuakeISNZ.m*.** uses Monte Carlo simulation with importance sampling of the magnitudes to simulate a set of earthquake scenarios. For all sources, a bin size of 0.1 is assumed for sampling from the magnitude distributions.

Call: M=QuakeISNZ(faults,backgrounds,distMchar)

Input parameters:

* *distMchar*. Indicates which recurrence relation distribution to use in sampling earthquake scenarios for fault sources (0=delta; 1=truncated normal)

Input variables: *faults*, *backgrounds* (both output by *loadingNZ.m*)

Output variables: *M.* Matrix including a row for each earthquake scenario generated, and the following 5 columns of information:

1. Scenario id number
2. ID of the background source or fault that generated the earthquake scenario
3. Magnitude of earthquake scenario
4. Indicator of whether it was generated by a background source (0) or a fault source (1)
5. Annual occurrence probability

**3.4.3. *SeisEventNZ.m, SeisEvent1NZ.m, SeisEvent3NZ.m***

*SeisEvent1NZ.m* and *SeisEvent3NZ.m* are alternative functions for simulating ground motions for each earthquake scenario. They both call the function *SeisEventNZ.m.*

***SeisEventNZ.m*** is the basic function that computes the ground motion intensity value at every site *i* given a specified earthquake scenario *j*. It does not consider spatial correlation of ground motion residuals, and it calculates only one ground motion intensity value for each earthquake scenario and each site. The functions *SeisEvent1NZ.m* and *SeisEvent3NZ.m* are based on and call this function. *SeisEventNZ.m* calls *GroundMotionNZ.m* (described in Section 3.4.9).

Call: [Event]=SeisEventNZ(inputfolder,M,faults,backgrounds,sites,Tsec)

Input parameters:

* + - * *inputfolder*. Location of folder with input files and .m files
      * *Tsec*. Period (seconds) of specified ground motion intensity parameter, *Sa(T)* (*Tsec*=0 refers to PGA)

Input variables:

* *faults* (output by *loadingNZ.m*)
* *backgrounds* (output by *loadingNZ.m*)
* *sites* (output by *loadingNZ.m*)
* *M* (output by *QuakeISNZ.m*)

Output variables: A structure called *Event* the size of which equals the number of earthquake scenarios. It contains:

* *Event(eq).year*: Year the earthquake occurred (or earthquake scenario ID if generated for earthquake scenarios based on importance sampling of magnitudes (*QuakeISNZ.m*))
* *Event(eq).SourceID*: Indicates which fault or area source was the source of the earthquake
* *Event(eq).rectype*: Indicator of whether it was generated by a background source (0) or a fault source (1) (same as 4th column of the M matrix)
* *Event(eq).M*: Magnitude of earthquake scenario
* *Event(eq).GM*: Four-term vector of ground motion intensity for every site location for a single earthquake scenario (in this order: mean, ln(mean), standard error of intra-event residual, standard error of inter-event residual)

***SeisEvent1NZ.m*** is the same as *SeisEventNZ.m* with two differences. First, it includes spatial correlation among the residuals if requested (*CorrCoef*=1). Second it simulates multiple ground motion maps for each earthquake scenario. The total number of ground motion maps generated from all the earthquake scenarios is determined by the input *num*, and they are distributed equally across earthquake scenarios. Note that in simulating ground motion maps, it truncates ground motions at +/- 3 standard deviations.

Call: [Event] = SeisEvent1NZ(inputfolder,M,faults,backgrounds,sites,CorrCoef,Tsec,num)

Input parameters:

* + - * *inputfolder*. Location of folder with input files and .m files
      * *Tsec*. Period (seconds) of specified ground motion intensity parameter, *Sa(T)* (*Tsec*=0 refers to PGA)
      * *CorrCoef*. If spatial correlation among residuals should be included (1); if not (0)
      * *num*. Total number of ground motion maps to be generated from all the earthquake scenarios

Input variables:

* *faults* (output by *loadingNZ.m*)
* *backgrounds* (output by *loadingNZ.m*)
* *sites* (output by *loadingNZ.m*)
* *M* (output by *QuakeISNZ.m*)

Output variables: A structure called *Event* that is similar to that output by *SeisEventNZ.m*, but is larger in size, includes total ground motion instead of mean in the .GM field, and includes annual occurrence probability:

* *Event(eq).year*: Year the earthquake occurred (or earthquake scenario ID if generated for earthquake scenarios based on importance sampling of magnitudes (*QuakeISNZ.m*))
* *Event(eq).SourceID*: Indicates which fault or area source was the source of the earthquake
* *Event(eq).rectype*: Indicator of whether it was generated by a background source (0) or a fault source (1) (same as 4th column of the M matrix)
* *Event(eq).M*: Magnitude of earthquake scenario
* *Event(eq).GM*: Four-term vector of ground motion intensity for every site location for a single earthquake scenario (in this order: total, ln(mean), standard error of intra-event residual, standard error of inter-event residual), where (total=mean+random intra-event residual+ random inter-event residual)
* *Event(eq).AP*: Annual occurrence probability

***SeisEvent3NZ.m*** is the same as *SeisEventNZ.m* with two differences. First, it includes spatial correlation among the residuals if requested (*CorrCoef*=1). Second it simulates multiple ground motion maps for each earthquake scenario. The total number of ground motion maps generated from all the earthquake scenarios is determined by the input *num*, and the number for each scenario is determined so as to minimize sampling variability (Section 2.4.4). *SeisEvent1NZ.m* is the same as *SeisEvent3NZ.m* except for the way they determine how to allocate the total number of requested ground motion maps among earthquake scenarios. Note that in simulating ground motion maps, it truncates ground motions at +/- 3 standard deviations.

Call: [Event] = SeisEvent3NZ (inputfolder,P,M,faults,backgrounds,sites,CorrCoef,Tsec,trueRGM,num)

Input parameters:

* + - * *inputfolder*. Location of folder with input files and .m files
      * *Tsec*. Period (seconds) of specified ground motion intensity parameter, *Sa(T)* (*Tsec*=0 refers to PGA)
      * *CorrCoef*. If spatial correlation among residuals should be included (1); if not (0)
      * *num*. Total number of ground motion maps to be generated from all the earthquake scenarios

Input variables:

* *faults* (output by *loadingNZ.m*)
* *backgrounds* (output by *loadingNZ.m*)
* *sites* (output by *loadingNZ.m*)
* *M* (output by *QuakeISNZ.m*)
* *P* (structure containing probabilities , i.e., probability of ground motion caused by earthquake scenario *j* at site *i* exceeds “true” ground motion at site *i* for return period *r*. It can be generated by running *PijrNZ.m*)
* *trueRGM* (can be output by *loadingNZ.m* or can be generated separately as explained in Section 3.1)

Output variables: A structure called *Event* that has the same structure as that output by *SeisEvent1NZ.m*

**3.4.4. *PijrNZ.m, PijrNZbin.m***

These functions compute the exceedance probability for each earthquake scenario (or ground motion map) *j*, site *i*, and return period *r*, where is the ground shaking from the “true” hazard curve for return period *r* at site *i*, and is the ground shaking at site *i* caused by earthquake scenario (or ground motion map) *j*.

***PijrNZ.m*** computes the exceedance probabilities for each earthquake scenarios *j*. The probabilities are between zero and one. It uses the Matlab function cdf() to numerically integrate over the distribution of ground motion intensity at site *i* in earthquake *j*, assuming a normal distribution truncated from -3 s.d. to + 3s.d. The output is used in *EQReduceNZ.m*.

Call: P=PijrNZ(M,sites,trueRGM,Event)

Input variables:

* *M* (output by *QuakeISNZ.m*)
* *sites* (output by *loadingNZ.m*)
* *trueRGM*. The “true” hazard curve data. Either provided as an input and generated with loadingNZ.m, or generated using MCS or PSHA (Figures 3 and 4).
* *Event* (output by *SeisEvent1NZ.m* or *SeisEvent3NZ.m*)

Output variables: A structure *P* which is the size of the number of earthquake scenarios *j* input. For every earthquake scenario *j*, it contains a matrix of with a row for each site *i* and a column for each return period.

***PijrNZbin.m*** computes the exceedance probabilities for each ground motion map *j*. They are binary, either zero or one, since there is no uncertainty in for a ground motion map. The output is used in the second implementation of the optimization model (Section 2.4.5).

Call: Pbin=PijrNZbin(Event,trueRGM)

Input variables:

* *trueRGM*. The “true” hazard curve data. Either provided as an input and generated with loadingNZ.m, or generated using MCS or PSHA (Figures 3 and 4).
* *Event* (output by *SeisEvent1NZ.m* or *SeisEvent3NZ.m*)

Output variables: A structure *Pbin* which is the size of the number of ground motion maps *j* input. For every ground motion map *j*, it contains a matrix of with a row for each site *i*, a column for each return period, and a value of zero or one in each cell.

**3.4.5. *EQReduceNZ.m***

This function takes a set of earthquake scenarios and selects the subset of those with the largest average contributions to the hazard curves (as defined in Equation 1), such that the cumulative contribution of the subset comprises a user-specified *contribpct* (Section 2.4.2). It also makes figures of the cumulative contribution vs number of earthquake scenarios (a) for all earthquake scenarios and (b) for only those selected earthquake scenarios.

Call: [RedM,RedP]=EQreduceNZ(M,P,contribpct)

Input parameters:

* *contribpct*. Cumulative percentage of contribution to hazard curves. *contribpct* is a value from 0 to 1. Determines how many earthquake scenarios will be retained in the reduced set. Due to the averaging across sites and return periods, the total cumulative percentage may exceed one by a small amount. Typically, this value will be 0.8 to 0.99.

Input variables:

* *M* (structure containing information about a set of earthquake scenarios; output by *QuakeISNZ.m*)
* *P* (structure containing probabilities , i.e., probability of ground motion caused by earthquake scenario *j* at site *i* exceeds “true” ground motion at site *i* for return period *r*. It can be generated by running *PijrNZ.m*)

Output variables:

* *RedM*. Reduced set of earthquake scenarios. Same format as *M*, but smaller in size since it only includes data for selected earthquake scenarios.
* *RedP*. Reduced set of exceedance probabilities. Same format as *P*, but smaller in size since it only includes data for selected earthquake scenarios.

**3.4.6. *UpdateMNZ.m and UpdateEventNZ.m***

These two functions take the output of the first and second implementations of the optimization, respectively, and update the set of earthquake scenarios or ground motion maps to reflect the results.

***UpdateMNZ.m*** uses the *Pj* values (i.e., adjusted annual occurrence probabilities for each earthquake scenario) output by the first implementation of the optimization to update the structures containing the set of earthquake scenarios and associated exceedance probabilities, . Values of *Pj*=0 indicate the corresponding earthquake scenario is not included in the reduced set. For those earthquake scenarios with *Pj*>0, the *Pj* becomes the new adjusted annual occurrence probability.

Call: [RedM2,RedP2]=UpdateMNZ(RedM1,RedP1)

Input variables:

* *RedM1*. Structure containing information about a set of earthquake scenarios (can be output by *EQReduceNZ.m*).
* *RedP1*. Structure containing probabilities , i.e., probability of ground motion caused by earthquake scenario *j* at site *i* exceeds “true” ground motion at site *i* for return period *r* (can be output by *EQReduceNZ.m*).

Input file

A tab-delimited text file containing a column of *Pj* values output by the optimization should be in the same folder as the Matlab function.

* *Pjs.txt*. Column vector of probabilities of each earthquake scenario *j*. First row is number of candidates in optimization run; second row is maximum allowable number of earthquake scenarios in the reduced set. These identify the optimization run (useful if more than one run is conducted). Remaining rows have *Pj* values for each candidate earthquake scenario *j*.

Output variables:

* *RedM2*. Reduced set of earthquake scenarios. Same format as *RedM1*, but smaller in size since it only includes data for selected earthquake scenarios.
* *RedP2*. Reduced set of exceedance probabilities. Same format as *RedP2*, but smaller in size since it only includes data for selected earthquake scenarios.

***UpdateEventNZ.m*** uses the *Pj* values (i.e., adjusted annual occurrence probabilities for each earthquake scenario) output by the second implementation of the optimization to update the structures containing the set of ground motion maps and associated exceedance probabilities, . Values of *Pj*=0 indicate the corresponding ground motion map is not included in the reduced set. For those ground motion maps with *Pj*>0, the *Pj* becomes the new adjusted annual occurrence probability.

Call: [RedEvent]=UpdateEventNZ(Event)

Input variables:

* *Event*. Structure containing information about a set of ground motion maps (can be output by *SeisEvent3NZ.m*).

Input file

A tab-delimited text file containing a column of *Pj* values output by the optimization should be in the same folder as the Matlab function.

* *Pjs2.txt*. Column vector of probabilities of each ground motion map *j*. First row is number of candidates in optimization run; second row is maximum allowable number of maps in the reduced set. These identify the optimization run (useful if more than one run is conducted). Remaining rows have *Pj* values for each candidate ground motion map *j*.

Output variables:

* *RedEvent*. Reduced set of ground motion maps. Same format as *Event*, but smaller in size since it only includes data for selected ground motion maps.

**3.4.7. *Summary1NZ.m, Summary2NZ.m***

The functions *Summary1NZ.m* and *Summary2NZ.m* are used to compute the summary results for a set of earthquake scenarios or a set of ground motion maps, respectively. Both provide a figure showing the histogram of hazard curve errors (HCEs) and the mean hazard curve error (MHCE), as defined in Han and Davidson (2012) and Section 3.5. Both also provide a table that can be used to generate maps of ground motion at each return period from the “true” and reduced set hazards. *Summary2NZ* also provides a figure showing the histogram of spatial correlation errors (SCEs) and the mean spatial correlation error (MSCE), as defined in Han and Davidson (2012) and Section 3.5. (*Summary1NZ* cannot provide the SCE results since spatial correlation can only be computed for a ground motion map, not an earthquake scenario.) The only other difference is that *Summary1NZ.m* calls *HazAnaResult1NZ.m* to compute the and *Summary2NZ.m* calls *HazAnaResult2NZ.m* to compute the .

Call: [HCE,Means,ToMap]=

Summary1NZ(Case,returnperiods,RedEvent,TrueEvent,TrueRGM)

Call: [HCE,SCE,Means,ToMap]=

Summary2NZ(Case,returnperiods,RedEvent,TrueEvent,TrueRGM)

Input parameters:

* + - * *Case*. Case identifies whether TrueRGM was provided or computed from a conventional MCS run.
        + Case='Given' means the “true” hazard is given. Input only: *RedEvent*, *TrueRGM*, and let *TrueEvent*=999.
        + Case='MCS' means the “true” hazard is from MCS. Input only: *RedEvent*, *TrueEvent*, and let *TrueRGM*=999
      * *returnperiods*. Vector of return periods (in years) at which to match to “true” hazard curves

Input variables:

Note that only two of the three variables should be input, as described in *Case*. They are as used in *HazAnaResult1NZ.m* and *HazAnaResult2NZ.m*.

* + - * *RedEvent*. Reduced set of ground motion maps (or lnmean and sigmas in case of earthquake scenarios)
      * *TrueEvent*. “true” set of ground motion maps if “true” hazard is determined by MCS
      * *TrueRGM*. “true” ground motions, . Matrix containing ground motion intensity value at each return period *r* (indicated in the inputs) and each site *i*. Includes a row for each site, and column for each return period.

Output variables:

* *HCE*. Hazard curve errors. . Matrix containing HCE value at each return period *r* (indicated in the inputs) and each site *i*. Includes a row for each site, and column for each return period.
* *SCE*. Spatial correlation errors.. Matrix containing SCE for each pair of sites. Includes a row and column for each site. (Only available for *Summary2NZ.m*.)
* *Means*. 1x2 vector. First value is MHCE; second is MSCE (Only available for *Summary2NZ.m*.).
* *ToMap*. Table of results to map return period ground motion maps. Includes a row for each site. First set of columns provides the ground motion for each return period from the reduced set; second set of columns provides “true” ground motion for each return period; third set of columns provides the difference, “true”-reduced.

**3.4.8. *PSHANZ.m***

This function performs a probabilistic seismic hazard analysis (PSHA) to compute the hazard curve at each specified site. It assumes ground motion follows a normal distribution truncated at +/- 3 standard deviations. It can be used for comparison with the MCS or OPS-based methods.

Call: [AEP,pshaRGM]=PSHANZ(inputfolder,backgrounds,faults,sites,Tsec,returnperiods)

Input parameters:

* + - * *inputfolder*. Location of folder with input files and .m files
      * *Tsec*. Period (seconds) of specified ground motion intensity parameter, *Sa(T)* (*Tsec*=0 refers to PGA)
      * *returnperiods*. Vector of return periods (in years) at which to match to “true” hazard curves

Input variables:

* *faults* (output by *loadingNZ.m*)
* *backgrounds* (output by *loadingNZ.m*)
* *sites* (output by *loadingNZ.m*)

Output variables:

* *AEP*. Matrix of hazard curve information for each site. First column contains the ground motion levels at which the hazard curve is defined (specified as *ybins* in one of the first few lines of code). Each remaining column contains the annual exceedence probability for a site *i* at the corresponding ground motion *y* in the first column. Thus there is a row for each ground motion intensity value at which the hazard curve is specified and (1+number of sites) columns.
* *pshaRGM*. Matrix containing ground motion intensity value at each return period *r* (indicated in the inputs) and each site *i*. Includes a row for each site, and column for each return period. This is if the results of this analysis are considered the “true” hazard.

**3.4.9. *GroundMotionNZ.m***

This function estimates the ground motion intensity in g at a single specified site caused by a single specified earthquake scenario (specifically, the mean, intra-event residual, and inter-event residual). If *Tsec* = 0, ground motion intensity is measured as PGA; otherwise it is spectral acceleration at period *Tsec*, *Sa(Tsec)*. The function uses the GMPEs in McVerry et al. (2006), with comments in the code identifying specific equations being used (Equations 1-6, 8-9). Figure 6 summarizes the main quantities computed, equations and coefficients used, and order of computation required.This function is called by the function *SeisEventNZ.m*.

Call: [lnGM,sigmaintra,tauinter,GM] =

GroundMotionNZ(inputfolder,Tsec,Mag,ftype,Rrup,dbottom,dtop,soiltype)

Input parameters:

* + - * *inputfolder*. Location of folder with input files and .m files
      * *Tsec*. Period (seconds) of specified ground motion intensity parameter, *Sa(T)* (*Tsec*=0 refers to PGA)

Input variables:

* *Mag*. (from *M*)
* *ftype*. (sensemov from *faults* or *backgrounds*)
* *Rrup*. (Rrup from *faults* or *backgrounds*, computed using *pfaultdistanceNZ.m* or *ppdistanceNZ.m*)
* *dbottom*. depth of bottom of fault (from *faults* or *backgrounds*)
* *dtop.* depth of top of fault (from *faults* or *backgrounds*)
* *soiltype.* soil type at site (from *faults* or *backgrounds*)

Output variables:

* *lnGM*. Natural log of mean ground motion intensity in g
* *sigmaintra*. Intra-event residual term for ground motion intensity in g
* *tauinter*. Inter-event residual term for ground motion intensity in g
* *GM*. Mean ground motion intensity in g

Notes:

* The program currently finds the stronger horizontal component of the horizontal components of ground motion (Table 5 in McVerry et al. 2006). If the geometric mean is desired, change the file name on line 48 to *McVerryT6.csv* so that it will read coefficients from Table 56 of McVerry et al. (2006) instead.
* For fault sources, centroid depth, *Hc*, is taken as middle point of top and bottom depth of fault; for background sources, it is the source depth given.
* The function assumes *r*=*Rhyp* for background seismicity sources and *r*=*Rrup* for fault sources. It computes *RTVZ* as the total distance *r* for sources of type nv (normal volcanic) and assumes *RTVZ*=0 for sources of type other than nv. As discussed in Section 6.1 of McVerry et al. (2006), the model assumes the hanging wall factor is always zero. Table 6 summarizes the choice of model and other input constants used for the GMPE.

Table 6. Summary of which McVerry et al. (2006) GMPE model/inputs to use for each source

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sense of movement** | **McVerry model** | **CN**  **(-1 or 0)** | **CR**  **(0.5, 1.0, or 0)** | **SI or DS?** |
| rv | Crustal | 0 | 1.0 | --- |
| sr, rs | Crustal | 0 | 0.5 | --- |
| nv, nn | Crustal | -1 | 0 | --- |
| ss | Crustal | 0 | 0 | --- |
| sn, ns | Crustal | 0 | 0 | --- |
| if | Subduction | --- | --- | SI |
| Depth ≥ 30 km and not “if” | Subduction | --- | --- | DS |

Basic steps in the function (Equations refer to McVerry et al. 2006)

1. Import coefficients file (from Table 5 or 6 McVerry et al. (2006))
2. Define *RVOL*: If fault type is “nv” then *RVOL*= *RRUP* else *RVOL*=0.
3. Centroid depth as average depth between top and bottom depth. For background seismicity, top and bottom depth will be the same.
4. Define *CN*, *CR*, *SI*, *DS*, *δc*, *δd* (see Table 6)
5. Identify if subduction model. All events classified as “if”, and those not classified as “if” but with depth greater than or equal to 30 km.
6. For subduction events:
   1. Get ln(PGAAB) and ln(PGA’AB) using Equation 2 and coefficients column 1 and 2, respectively.
   2. If soil type C, D, or E:

* If T=0 then, use Equation 4 to compute PGAC/D.
* else:
  + - Use Equation 4 to compute ln(PGAC/D) and ln(PGA’C/D);
    - Use Equation 2 to compute ln(SA’AB)
    - Use Equation 4 to compute ln(SA’C/D)
    - Use Equation 6 to compute SAC/D
  1. If soil type A or B do:
* If T=0, compute PGAAB
* If T>0:
  + - Use Equation 2 to compute ln(SA’AB)
    - Use Equation 6 to compute SAAB.

1. For crustal events:
   1. Get ln(PGAAB) and ln(PGA’AB) using Equation 1 and coefficients column 1 and 2, respectively.
   2. If soil type C or D:
   * If T=0 then, use Equation 4 to compute PGAC/D.
   * Else:
     + - Use Equation 4 to compute ln(PGAC/D) and ln(PGA’C/D)
       - Use Equation 1 to compute ln(SA’AB)
       - Use Equation 4 to compute ln(SA’C/D)
       - Use Equation 6 to compute SAC/D
   1. If soil type A or B:

* If T=0, compute PGAAB
* If T>0:
  + - Use Equation 1 to compute ln(SA’AB)
    - Use Equation 6 to compute SAAB.

1. Compute σ from Equation 8 and obtain τ from coefficients.

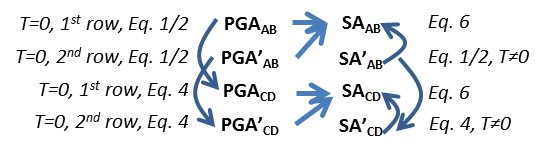


Figure 6. Summary of quantities computed in McVerry et al. (2006) GMPEs, with equations and coefficients used for each. Arrows indicate which quantities are required to compute other quantities (e.g., PGACD requires PGAAB as input)

**3.4.10. *ppdistanceNZ.m*** and ***pfaultdistanceNZ.m***

These functions compute point-to-point distances and point-to-fault distances, respectively. For faults, the distance is *Rrup*, the closest distance to the rupture plane. The functions use the Haversine formula to compute the distance between two points whose locations are defined in terms of longitude and latitude (http://www.movable-type.co.uk/scripts/latlong.html; http://www.codeguru.com/cpp/cpp/algorithms/article.php/c5115/Geographic-Distance-and-Azimuth-Calculations.htm). The function *pfaultdistanceNZ.m* follows the method in Kaklamanos et al. 2011. It first computes RJB and the source-to-site azimuth, then uses the equations in that paper to compute *Rrup*. These functions are called by *loadingNZ.m*.

Call: [ distance ] = ppdistanceNZ(lat1,lon1,dep1,lat2,lon2,dep2 )

Inputs: latitude and longitude (in decimal degrees), and depth (km) for each point

Call: [ Rrup ] = pfaultdistanceNZ( siteLAT,siteLON,ID,faults )

Inputs: latitude and longitude of site (in decimal degrees), id of fault being considered, *faults* (output by *loadingNZ.m*)

**3.4.11. *spatialcorre.m***

This function calculates the spatial correlation coefficient of ground motion values at every pair of sites (*a*,*b*) in the set of ground motion maps. It is computed using Eq. 13 (Section 3.5).

Call: cc=spatialcorre(Event);

Input variables:

* *Event* (output by *SeisEvent1NZ.m* or *SeisEvent3NZ.m*)

Output variables:

* *cc*. Upper triangular matrix of spatial correlation coefficient of every two sites.

**3.4.12. *HazAnaResult1NZ.m and HazAnaResult2NZ.m***

*HazAnaResult1NZ.m* calculates hazard curves for a set of earthquake scenarios, either those generated by MCS and representing the “true” hazard, or those from a reduced set of earthquake scenarios. *HazAnaResult2NZ.m* calculates hazard curves for a set of ground motion maps, either those generated by MCS and representing the “true” hazard, or those from a reduced set of ground motion maps. Both require the same input, but *HazAnaResult1NZ.m* integrates over the ground motion distribution to compute the exceedence probability and *HazAnaResult2NZ.m* sums the probabilities of all ground motion maps that exceed a specified ground motion intensity. In the case of *HazAnaResult1NZ*, only the lnmean and sigmas are being used from the input *Event*.

Call: [HCD,RGM]=HazAnaResult1NZ(Event, returnperiods, trueRGM)

Call: [HCD,RGM]=HazAnaResult2NZ(Event,returnperiods, trueRGM)

Input parameters:

* *returnperiods*. Vector of return periods (in years) at which to match to “true” hazard curves

Input variables:

* *Event*. Set of ground motion maps, either reduced set or “true” if generated from MCS. For *HazAnaResult1NZ*, Event should be generated from the set of earthquake scenarios *M* using a run of *SeisEvent1NZ.m* with number of eqs same as number of eqs in *M*. This will provide the lnmean and sigmas necessary to integrate over the ground motion distribution).
* *trueRGM*. The “true” hazard curve data. Either provided as an input and generated with loadingNZ.m, or generated using MCS or PSHA (Figures 3 and 4).

Output variables:

* *HCD*. Matrix containing annual exceeding probability of specific ground motion value at every site *i*. Includes a row for each site, and column for each ground motion value. A ground motion range [0.001:0.001:3] (in *g*) is considered in the code, that is, the annual exceeding probability of every ground motion value in the range [0.001:0.001:3] is calculated at every site.
* *RGM*. Matrix containing ground motion intensity value at each return period *r* (indicated in the inputs) and each site *i*. Includes a row for each site, and column for each return period. This is if the scenarios are the “true” hazard generated from MCS, or if they are estimated values from a reduced set.

**3.4.13. *DataGeneration.py***

This Python function reads in exceedance probability data,, from a .mat format and saves them into a dictionary that is readable by Python for use in the *SolveMe.py* function. It can read multiple P matrices in one run.

There is no call for this function. It can be run by opening it in the editor and clicking the Run button.

Inputs:

Inputs are provided by changing values in the code as follows:

* Line 18. Return period values in years
* Line 19. Name of matlab variable(s) that contains the *P* matrix with the exceedance probabilities , in same format as *P* and *RedP* matrices defined in matlab functions, such as, Section 3.4.4. Do not include the file extension .mat. Multiple variables can be listed if desired.
* Line 20. Number of candidate earthquake scenarios or ground motion maps. Multiple values can be listed, but they should be in the same order as the corresponding exceedance probability files in Line 19. The matlab variable in Line 19 should be the same size as indicated here.
* Line 21. Number of sites
* Line 22. Path of .mat file that contains the matlab variable(s) in Line 19. If it is in the same folder as this function, just the file name is required.

Outputs:

Binary file with name given in Line 7 that contains the P matrix values but in Python format. The user never needs to look at this. It will simply be input into *SolveMe.py*.

**3.4.14. *SolveMe.py***

This Python function creates an optimization model that can be used to select a reduced set (i.e., subset) of earthquake scenarios or ground motion maps and assign adjusted occurrence probabilities using the OPS method (Sections 2.4.3 and 2.4.5). If applied for earthquake scenarios, the exceedance probabilities are between zero and one; if applied for ground motion maps, they are binary, either zero or one. Otherwise, everything is the same in the two applications. The function assumes objective function weights of *wir*=*r* (Section 2.4.3). The outputs of the model are the objective function value and the model decision variables: the hazard-consistent annual occurrence probability for each earthquake scenario or ground motion map (*Pj*) and hazard curve errors for each site *i* and return period *r* ( and ). The function also outputs the binary variables that indicate whether or not earthquake scenario *j* is included in the selected reduced set of events. It is not really needed as output, since it is just one for all *Pj*>0 and zero otherwise. Multiple runs of the optimization can be conducted in a one call to this function. Since the second implementation of the optimization sometimes takes a long time to solve, the user can specify a maximum time limit for letting the function run. The solution is saved in a Python dictionary.

There is no call for this function. It can be run by opening it in the editor and clicking the Run button.

Input parameters:

Inputs are provided by changing values in the code as follows:

* Line 18. Return period values in years
* Line 19. Name of matlab variable(s) that contains the *P* matrix with the exceedance probabilities , in same format as *P* and *RedP* matrices defined in matlab functions, such as, Section 3.4.4. Do not include the file extension .mat. Multiple variables can be listed if desired.
* Line 20. Number of candidate earthquake scenarios or ground motion maps. Multiple values can be listed, but they should be in the same order as the corresponding exceedance probability files in Line 19. The matlab variable in Line 19 should be the same size as indicated here.
* Line 21. Value(s) of maximum allowable number of earthquake scenarios (or ground motion maps) in reduced set, *Jred.* Multiple values can be listed.
* Line 22. Number of sites
* Line 23. Maximum time to allow optimization solver to run in seconds.
* Line 100. Specify solver to use, either “GUROBI” or “PULP\_CBC\_CMD” (without quotation marks).

Output variables:

Binary file with name given in Line 19 that contains:

* Objective function value
* Hazard-consistent annual occurrence probability for each earthquake scenario or ground motion map (*Pj*)
* Hazard curve errors for each site *i* and return period *r* ( and )
* Binary variables that indicate whether or not earthquake scenario (or ground motion map) *j* is included in the selected reduced set (*zj*)

The user never needs to look at this. It will simply be input into *Print.py*.

**3.4.15. *Print.py***

This Python function converts the output from *SolveMe.py* into an Excel format more easily readable by the user.

There is no call for this function. It can be run by opening it in the editor and clicking the Run button.

Input parameters:

Inputs are provided by changing values in the code as follows:

* Line 20. Number of candidate earthquake scenarios (or ground motion maps). Multiple values can be provided.
* Line 21. Value(s) of maximum allowable number of earthquake scenarios (or ground motion maps) in reduced set, *Jred.* Multiple values can be listed.
* Line 22. Number of sites

To format the output, it is also necessary to type in the Number of candidates and Jred values in section (iii) of the code for each of these three output files:

* Pjs.txt
* Zjs.txt
* Obj.txt

Output variables (in text file format):

In the following, a run refers to an optimization formulation with a specified combination of number of candidates and number of allowable earthquake scenarios (or ground motion maps) in the reduced set (*Jred*). A single call to *SolveMe.py* can solve multiple runs.

* *Pjs.txt*. There is a column of data for each run. In each column, the first row contains the number of candidates, the second row contains the *Jred* value, and the remaining rows contain the *Pj* values for each candidate earthquake or ground motion map from *j*=1 to the number of candidates.
* *Zjs.txt*. There is a column of data for each run conducted. In each column, the first row contains the number of candidates, the second row contains the *Jred* value, and the remaining rows contain the *Zj* values for each candidate earthquake scenario or ground motion map from *j*=1 to the number of candidates. All *Zj* values should be 0 or 1.
* *Obj.txt*. There is a column of data for each run conducted. In each column, the first row contains the number of candidates, the second row contains the *Jred* value, and the third contains the final objective function value.
* *errors\_X\_Y.txt*. There is an *errors\_X\_Y.txt* file for each run. In the file name, X refers to the number of candidates and Y refers to the number of allowable earthquake scenarios (or ground motion maps) in the reduced set (*Jred*). Each file contains a matrix with a row for each site *i* and a column for each return period *r*. The value in each cell is the error ( or ) for the corresponding site and return period. In the matrix, are presented as positive values; are presented as negative values. Since both and cannot be positive for a specified *i* and *r*, they are combined into this single output file.

**3.5. Output files/Results**

The main results of the method are:

* **Reduced set of earthquake scenarios with associated annual occurrence probabilities**

This result is stored in the variable *RedM2*, which includes scenario ID, ID of source that generated it, magnitude, indicator of if it was a background or fault source, annual occurrence probability (see Sections 3.4.2 and 3.4.7).

* **Reduced set of ground motion maps with associated annual occurrence probabilities**

This result is stored in the variable *RedEvent*, which includes map ID, ID of source that generated it, indicator of if it was a background or fault source, magnitude of earthquake scenario that generated it, for every site location a vector of ground motion intensity that includes (total, ln(mean), intra-event residual, inter-event residual), annual occurrence probability (see Sections 3.4.3 and 3.4.7).

Additional outputs that can be used to assess the quality of the results are provided as output from *Summary1NZ.m* or *Summary2NZ.m* (Section 3.4.9). They include:

* **Hazard curve errors for every site *i* and return period *r***, *HCE* (defined in Eq. 12).

Can be used to plot a histogram of the HCEs (as in Fig. 7 of Han and Davidson 2012)

* **Mean hazard curve error,** *MHCE* (defined in Eq. 13).
* **Spatial correlation errors for every pair of sites *a* and *b****, SCE* (defined in Eq. 15).

Can be used to plot a histogram of the HCEs (as in Fig. 7 of Han and Davidson 2012).

* **Mean spatial correlation error**, *MSCE* (defined in Eq. 14).

*HCE* is the error in the hazard curve in terms of ground motion intensity (e.g., PGA) as % of the “true” value, i.e., the horizontal distance from the reduced set hazard curve to the “true” hazard curve for site *i* and return period *r* (Figure 1) divided by the true ground motion at site *i* with return period *r* (Eq. 12):

(12)

(13)

where is the “true” ground motion intensity at site *i* and return period *r*, is the ground motion intensity at site *i* and return period *r* based on the hazard curves from the reduced set.

The spatial correlation pattern for the study region is represented by a matrix of weighted correlation coefficients, where the weights are the annual ground motion map occurrence probabilities, *Pn*. For each pair of sites *a* and *b*, where *yan* and *ybn* are, respectively, the ground motions at sites *a* and *b* in ground motion map *n*, the weighted correlation coefficient *ρa,b* is:

(14)

The spatial correlation error (*SCE*) for sites *a* and *b* is defined as:

(15)

where and are the weighted correlation coefficients calculated in Equation 16 using, respectively, the full set of 100,000 years of “true” MCS ground motion maps, or the reduced set of probabilistic ground motion maps. A summary measure of the overall spatial correlation match then, is the mean, over all pairs of sites, of the absolute value of *SCE* (Eq. 16). Note that SCE and MSCE can only be calculated if the “true” hazard curves are based on conventional MCS, not given (Section 3.1).

(16)

**3.6. Conducting multiple runs**

It may be useful to use multiple parameter values (e.g., *Jred*) at each step to ensure the best solution is obtained (see Appendix C, Section 9.2 for guidance on implementation of the process). This section contains a few notes on how to do that at each step in the process.

First, generate candidate earthquake scenarios using importance sampling in Step 1.

Step 2. To consider multiple values of cumulative contribution (and thus get multiple sets of earthquake scenarios for use in the first implementation of the optimization), set the highest value you want to use in EQReduceNZ.m. In the optimization in Step 3, one can then set multiple values of number of candidate scenarios to consider (Njc) and the optimization will use only the highest contributors in each case. For example, if 99% results in 6500 candidate earthquakes, then one requests candidate sizes of 6500 and 4000, the optimization will first do a run with all 6500, then do a run with the top 4000 contributors.

Step 3. To consider multiple values of number of earthquake scenarios allowed in the reduced set (Njr or *Jred*), simply input multiple values in *SolveMe.py* in the first implementation of the optimization. You can also input multiple candidate sets of earthquake scenarios by inputting a P matrix for each.

Step 4. To consider multiple numbers of candidate ground motion maps (Nmc), repeat Steps 3b, 4, 5a for each run. Alternatively, you can use the function sensNjrTONmcNZ.m to loop through the runs automatically.

Step 5. To consider multiple values of number of ground motion maps allowed in the reduced set (Nmr or *Jred*), simply input multiple values in *SolveMe.py* in the second implementation of the optimization. You can also input multiple candidate sets of ground motion maps by inputting a P matrix for each.

**4. Matlab/Python commands to implement method**

In all cases, these commands assume we consider sources within 200 km, use PGA, no spatial correlation in the ground motion residuals, and return periods of 250, 500, 1000, and 2500 years. The input folder should describe where all input and function files are. All commands are in Matlab except where noted to use Python.

**4.1. OPS method to generate reduced set of earthquake scenarios only**

These commands assume a delta function for the fault recurrence function and that the user wants to keep the earthquake scenarios that together make up 99% of the hazard contribution.

inputfolder='C:\Users\RachelD\Documents\New Zealand\GNS hazard project\For Nick\Code\'

[ faults,backgrounds,sites, trueRGM ] = *loadingNZ*( inputfolder,200 );

M=QuakeISNZ(faults,backgrounds,0);

Suppose this produces a structure M with 73,858 earthquake scenarios...

[Event1]=*SeisEvent1NZ*(inputfolder,M,faults,backgrounds,sites,0,0, 73858);

P=*PijrNZ*(M,sites,trueRGM,Event1);

[RedM1,RedP1]=*EQreduceNZ*(M,P,0.99);

------------------

In Python:

Put RedP1.mat in same folder as *DataGeneration.py*, *SolveMe.py*, and *Print.py*.

In *DataGeneration.py*, set the input parameters as indicated in Section 3.4.13 and run.

In *SolveMe.py*, set the input parameters as indicated in Section 3.4.14 and run.

In *Print.py*, set the input parameters as indicated in Section 3.4.15 and run:

------------------

Put Pjs.txt, output from *Print.py*, into folder with Matlab functions

[RedM2,RedP2]=*UpdateMNZ*(RedM1,RedP1);

The final set of earthquake scenarios is stored in RedM2. Next two commands can be used to compute errors associated with that final set of earthquake scenarios, assuming “true” hazard is given from PSHA. Suppose there are 102 earthquake scenarios in the RedM2 set…

[EventTemp]=*SeisEvent1NZ*(inputfolder,M,faults,backgrounds,sites,0,0, 102);

returnperiods=[250 500 1000 2500];

[HCE,Means,ToMap]=*Summary1NZ*(‘Given’,returnperiods, EventTemp,999,trueRGM);

**4.2. OPS method to generate reduced set of ground motion maps**

Same as commands in Section 4.1 through *UpdateMNZ.m*. Then continue with the following commands, which assume the user wants to generate 3000 candidate ground motion maps.

[Event2] = *SeisEvent3NZ*(inputfolder,RedP2,RedM2,faults,backgrounds,sites,0,0, trueRGM,3000);

Pbin=*PijrNZbin*(Event2,trueRGM);

------------------

In Python:

Put Pbin.mat in same folder as *DataGeneration.py*, *SolveMe.py*, and *Print.py*.

In *DataGeneration.py*, set the input parameters as indicated in Section 3.4.13 and run.

In *SolveMe.py*, set the input parameters as indicated in Section 3.4.14 and run.

In *Print.py*, set the input parameters as indicated in Section 3.4.15 and run:

------------------

Put Pjs2.txt, output from *Print.py*, into folder with Matlab functions

[RedEvent]=*UpdateEventNZ*(Event2);

[HCE,SCE,Means,ToMap]=*Summary2NZ*(‘Given’,returnperiods,RedEvent,999, trueRGM);

**4.3. Conventional MCS**

These commands assume simulation for 1,000,000 years, and a delta function for the fault recurrence function.

inputfolder='C:\Users\RachelD\Documents\New Zealand\GNS hazard project\For Nick\Code\'

[ faults,backgrounds,sites, trueRGM ] = *loadingNZ*( inputfolder,200 );

[M]=*QuakeNZ*(1000000,faults,backgrounds,0);

Suppose this produces a structure M with 698,839 earthquake scenarios...

[Event]=*SeisEvent1NZ*(inputfolder,M,faults,backgrounds,sites,0,0, 698839);

**4.4. PSHA**

inputfolder='C:\Users\RachelD\Documents\New Zealand\GNS hazard project\For Nick\Code\'

[ faults,backgrounds,sites, trueRGM ] = loadingNZ( inputfolder,200 );

returnperiods=[250 500 1000 2500];

[AEP,pshaRGM]=PSHANZ(inputfolder,backgrounds,faults,sites,0,returnperiods);

**5. Potential future extensions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **Current version** | **Possible future modification** | **Required changes** |
| GMPE equations and soil site effects | McVerry et al. (2006) | Allow weighted average of McVerry and Bradley models | Develop input files that contain any additional information required by Bradley models (e.g., *Vs30*). Rewrite *GroundMotionNZ.m* to implement Bradley model |
| Weights for objective function in OPS Steps 2 and 4 (Section 2.4.2) | All *wir*=*r* | Allow alternative user-specified weights for specified locations or return periods | Modify *SolveMe.py* function to specify different weights |
| Distributed seismicity sources | Treat as point sources | Allow finite fault geometries | Modify *GroundMotionNZ.m*. Use Stirling et al. (2012, Eqns. 1-3) to get a rupture area or length for specified magnitude. Randomly sample azimuth. |
| Distributed seismicity sources recurrence relation | Not time-dependent | Allow time-dependent (TD). | Modify optimization formulation in *SolveMe.py* and associated input and other files to incorporate time. |

**References** (and what they were used for)

Journal papers

Han, Y., and Davidson, R. 2012. Probabilistic seismic hazard analysis for spatially distributed infrastructure. Earthquake Engineering and Structural Dynamics 41(15), 2141-2158.

*Description of method being implemented, including comparison to other possible methods and empirical example for Los Angeles.*

Jayaram N, Baker J. Correlation model for spatially distributed ground-motion intensities. Earthquake Engineering and Structural Dynamics 2009; 38:1687–1708.

*Model to estimate spatial correlation in ground motion residuals.*

Kaklamanos, James, Laurie G. Baise, David M. Boore (*2011*) Estimating Unknown Input Parameters when Implementing the NGA Ground-Motion Prediction Equations in Engineering Practice. Earthquake Spectra: November 2011, Vol. 27, No. 4, pp. 1219-1235.

*Guidelines for estimating RRUP (and other GMPE inputs)*

McVerry, G.H.; Zhao, J.X.; Abrahamson, N.A.; Somerville, P.G. 2006 New Zealand acceleration response spectrum attenuation relations for crustal and subduction zone earthquakes. *Bulletin of the New Zealand Society for Earthquake Engineering, 39(1):*1-58

*New Zealand ground motion prediction equation model currently implemented in program.*

Stirling, M.W.; McVerry, G.H.; Berryman, K.R. 2002 A new seismic hazard model for New Zealand. *Bulletin of the Seismological Society of America, 92(5):*1878-1903

*Parameter a, Gutenberg-Richter relation*

Stirling, M.W.; McVerry, G.H.; Gerstenberger, M.C.; Litchfield, N.J.; Van Dissen, R.J.; Berryman, K.R.; Barnes, P.; Wallace, L.M.; Villamor, P.; Langridge, R.M.; Lamarche, G.; Nodder, S.; Reyners, M.E.; Bradley, B.; Rhoades, D.A.; Smith, W.D.; Nicol, A.; Pettinga, J.; Clark, K.J.; Jacobs, K. 2012 National seismic hazard model for New Zealand : 2010 update. *Bulletin of the Seismological Society of America, 102(4):*1514-1542; doi:10.1785/0120110170

*Includes relationship between magnitude and rupture area or length. Not used in current version, but could be used if program was modified to allow finite fault geometry for distributed seismicity sources or if fault sources were modified to allow less than the entire fault rupturing.*

Websites

Websites used to compute distances and angles

<http://www.movable-type.co.uk/scripts/latlong.html>

<http://www.codeguru.com/cpp/cpp/algorithms/article.php/c5115/Geographic-Distance-and-Azimuth-Calculations.htm>

*Haversine formula and Law of cosines in spherical coordinates, online references.*

**APPENDIX A**

**Python and Gurobi software installation notes**

## Python

We used Python for the optimization steps. While there are many high-level languages, Python is one of the easiest languages to learn and use, while at the same time being very elegant. Go to the Downloads page (<http://www.python.org>) and follow the instructions based upon your operating system. Be careful to choose the version for your operating system and hardware. We used Python 2.7.6 and the Wing IDE editor (free and professional versions available from <http://wingware.com/>). A later version or different editor is acceptable, but ***please do not use Python 3*** as it is quite different.

Five main python libraries are needed to run the code:

1. **pickle** to make a binary file from output
2. **Scipy** for general computation
3. **Numpy** for general computation
4. **PuLP** to create the model
5. **gurobipy** to solve the model (only if using the Gurobi solver)

**pickle**

The pickle module implements a fundamental, but powerful algorithm for serializing and de-serializing a Python object structure. You do not need to install it as it comes with Python.

**scipy, and numpy, and PuLP**

NumPy and SciPy are the fundamental packages for scientific computing with Python. PuLP is a free open source software written in Python. It is used to describe optimization problems as mathematical models. PuLP can then call any of numerous external Linear Programming solvers (e.g., CBC, GLPK, CPLEX, Gurobi) to solve this model and then use python commands to manipulate and display the solution. By installing PuLP, you will immediately have access to PULP\_CBC\_CMD solver that we used for optimization model. To obtain more information about PuLP we suggest the following link:

<http://www.optimization-online.org/DB_FILE/2011/09/3178.pdf>.

To install all three of these libraries on a windows machine, go to this link: <http://www.lfd.uci.edu/~gohlke/pythonlibs/>. Search for the name of the library (they are listed by date, not alphabetically), and choose the correct version (for Windows 32-bit or 64-bit; and compatible with Python 2.7). Download the executable to install.

Alternatively, to install PuLP, you can also visit <https://pythonhosted.org/PuLP/>. You can also find the latest version of PuLP here <https://pypi.python.org/pypi/PuLP>.

**gurobipy**

The Gurobi Optimizer is a state-of-the-art solver for mathematical programming. It can be called via many programming languages including Python, C++, and MATLAB. We used Python. Gurobi is a solver that is free for academics, but not free for others.To know more about Gurobi Python interface, the interested reader can check <http://www.gurobi.com/resources/seminars-and-videos/modeling-with-the-gurobi-python-interface>.

To install Gurobi, go to this link:

<http://www.gurobi.com/documentation/6.0/windows-quick-start-guide/software_installation_guid>

After you download gurobi, follow instructions to install. You will have to download a license as well. You can find pricing information on the website. There are free trials available.

**APPENDIX B**

**Soil site classifications**



**APPENDIX C.**

**Christchurch case study results**

**Table of Contents**

1. Introduction
2. Comparison of base results

2.1. Mean hazard curve errors

2.2. Histograms of hazard curve errors (HCEs) for each comparison

1. Monte Carlo simulation results
2. Importance sampling results (Step 1)
3. Reduction in importance sampling results based on contribution to hazard curves results (Step 2)
4. Reducing earthquake scenarios using optimization (Step 3)
5. Generating candidate ground motion maps (Step 4)
6. Reducing ground motion maps using optimization (Step 5)
7. Conclusions
   1. Final recommended reduced set of ground motion maps
   2. Conclusions and recommendations for applying method

**1. Introduction**

This document summarizes results of applying the modified Han and Davidson (2012) method. A sensitivity analyses were conducted for each of the five steps in the analysis, and the results are presented.

Notes about scope and input

* Christchurch, NZ study area is defined by a rectangle from (-43.7°, 172.3°) to (-43.3°, 172.9°). It is divided into 35 0.1° by 0.1° grid cells. (Green rectangle in Fig. 1)
* All runs include only fault and background sources with *Rrup* ≤ 200 km (Fig. 1)
* Ground motion is truncated at +/- 3 s.d.
* No spatial correlation among residuals
* *Mmin* =5.25 for background sources

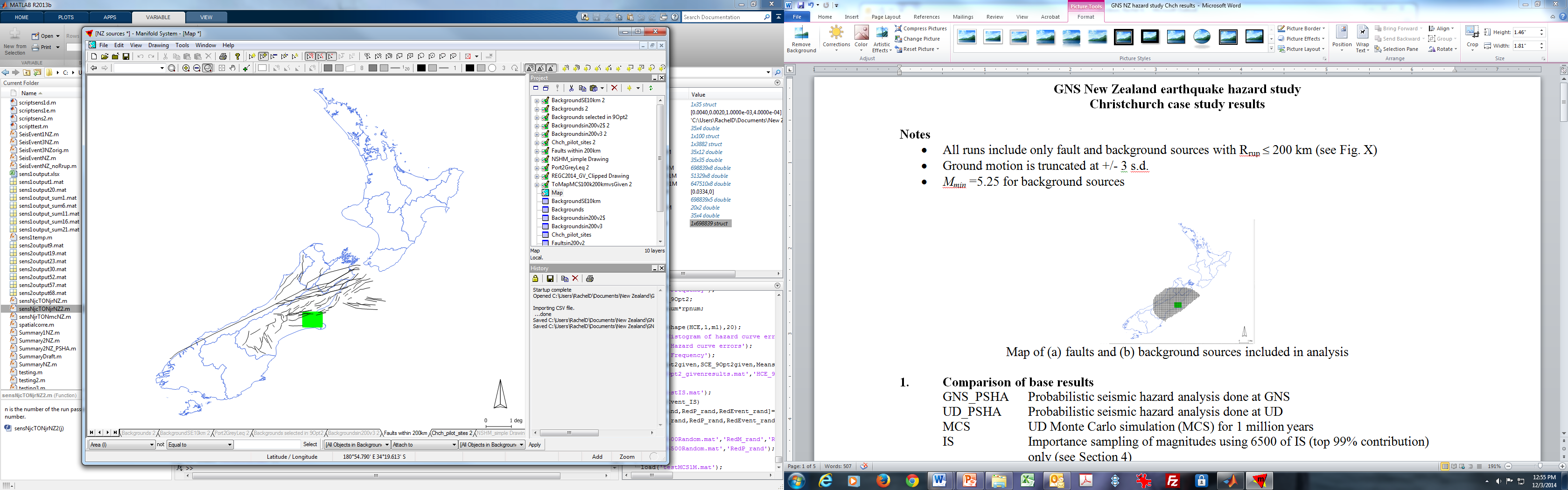
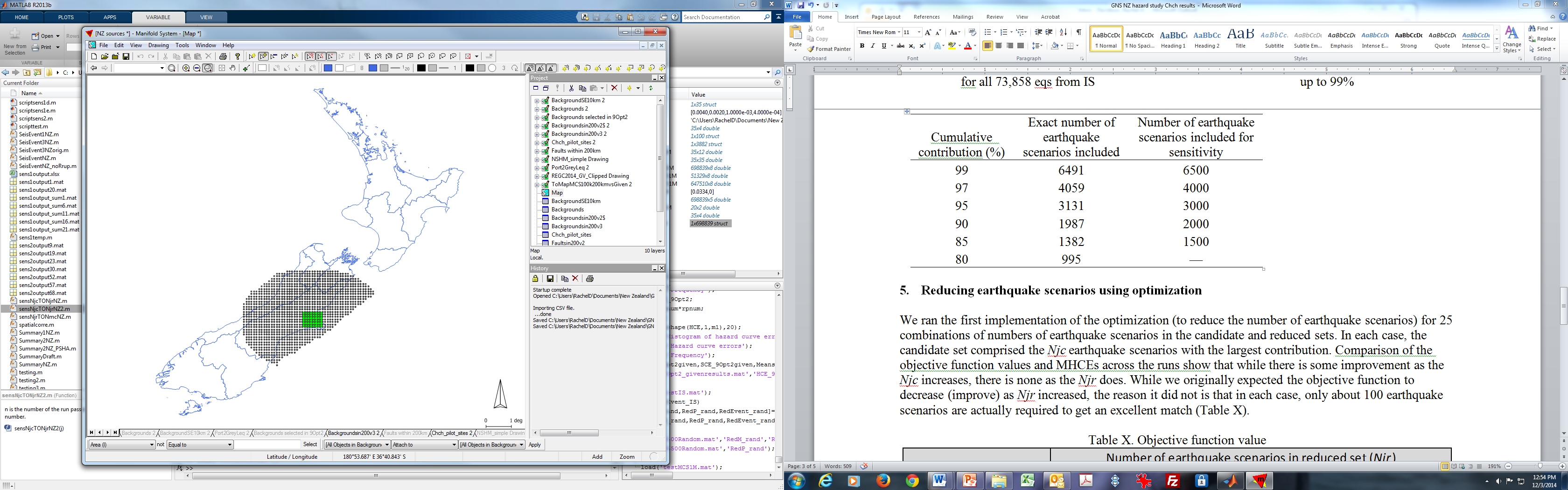
 

Figure 1. Maps of (a) faults and (b) background sources included in analysis (within 200 km of Chch)

**2. Comparison of base results**

GNS\_PSHA Probabilistic seismic hazard analysis done at GNS

UD\_PSHA Probabilistic seismic hazard analysis done at UD

MCS UD Monte Carlo simulation (MCS) for 1 million years

IS Importance sampling of magnitudes using 6500 of IS (top 99% contribution) only (see Section 4)

Comparisons show all runs match quite well, so that it is reasonable to compare final OPS-based method results to GNS\_PSHA and MCS (for checking spatial correlation)

**2.1. Mean hazard curve errors**

As defined in Han and Davidson (2012), hazard curve error (*HCE*) is the error in the hazard curve in terms of *PGA* as % of the “true” value (Eq. 1), i.e., the horizontal distance from the reduced set hazard curve to the “true” hazard curve for site *i* and return period *r* divided by the true ground motion at site *i* with return period *r* (Fig. 2). A summary measure of the overall hazard curve match then, is the mean over all sites and return periods, of the absolute value of *HCE* (Eq. 2).

(1)

(2)

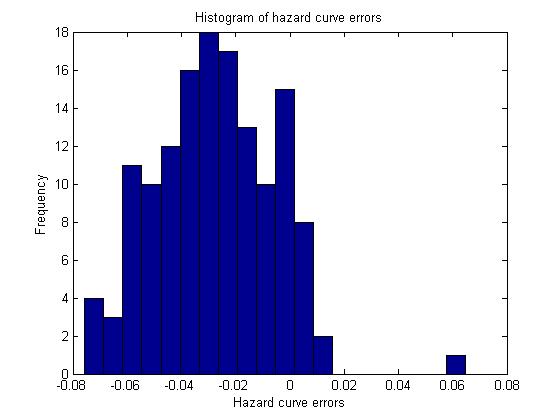
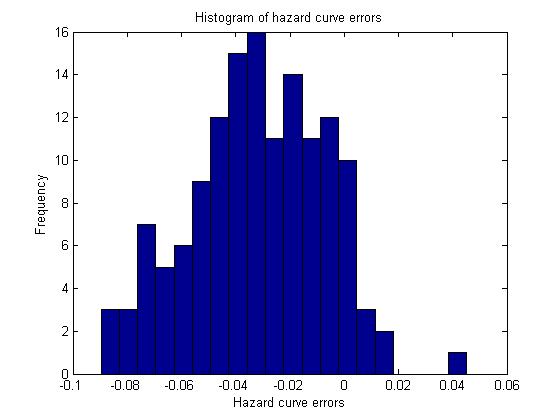


Fig. 2. Schematic defining errors between the “true” and reduced set hazard curves for site *i*

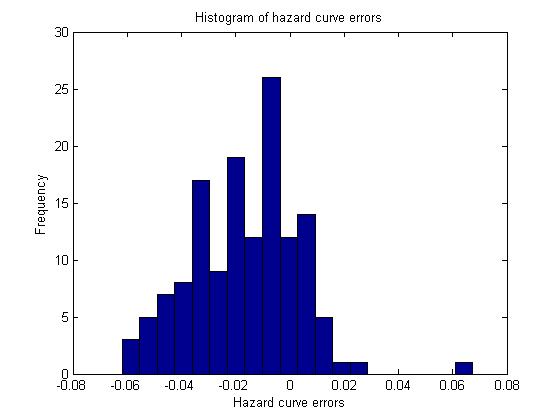
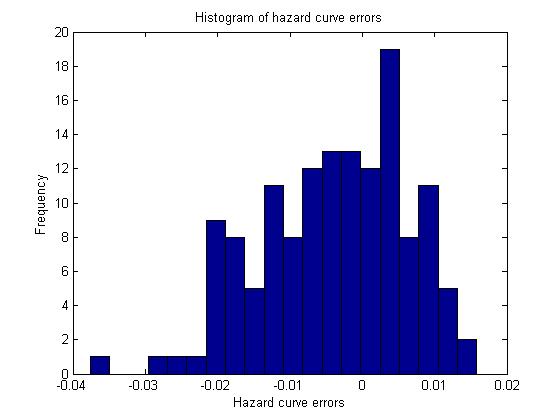
Table 1. Mean hazard curve errors (MHCEs)

|  |  |  |  |
| --- | --- | --- | --- |
|  | UD\_PSHA | MCS | IS |
| GNS\_PSHA (true) | 0.0295 | 0.0334 | 0.0205 |
| UD\_PSHA (true) |  | 0.0087 | 0.0106 |
| MCSb (true) |  |  | 0.0151 |

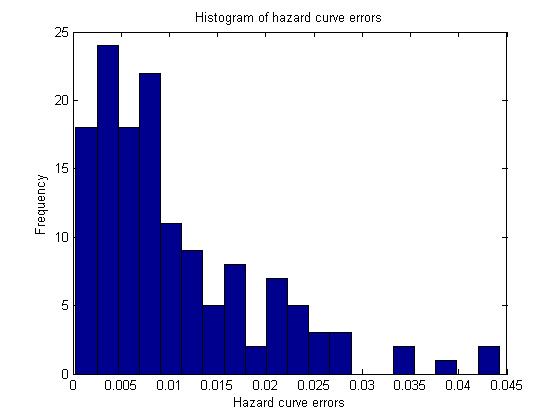
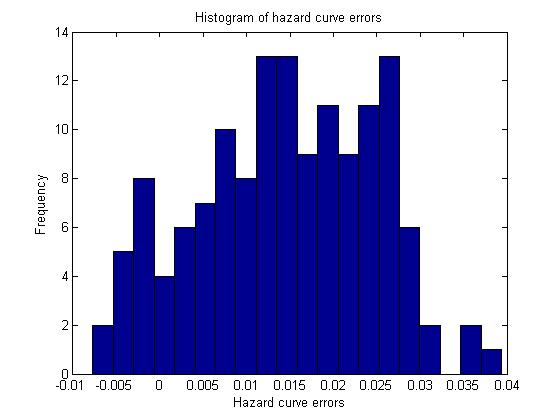
**2.2. Histograms of hazard curve errors (HCEs) for each comparison**

UD PSHA vs GNS PSHA (true) MCS vs. GNS PSHA (true)

IS vs. GNS PSHA (true) MCS vs. UD PSHA (true)

IS vs. UD PSHA (true) IS vs. MCS (true)

Figure 3. Histograms comparing hazard curve errors (HCEs) for six pairwise comparisons listed in Table 1

**3. Monte Carlo simulation results**

MCS for 1 million years resulted in 698,839 earthquake scenarios. The error continued to decline up to at least 1 million years (Fig. 4), so we used the results for that time period.

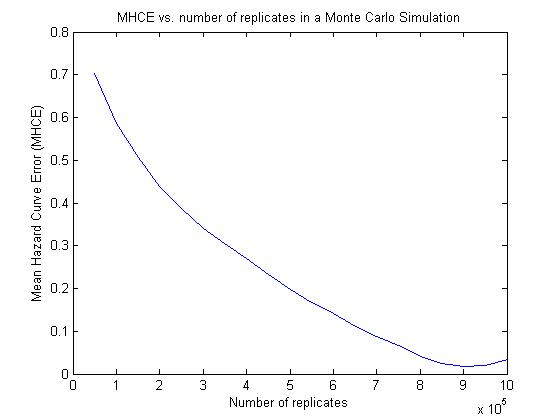


Figure 4. MHCE vs number of replicates for MCS

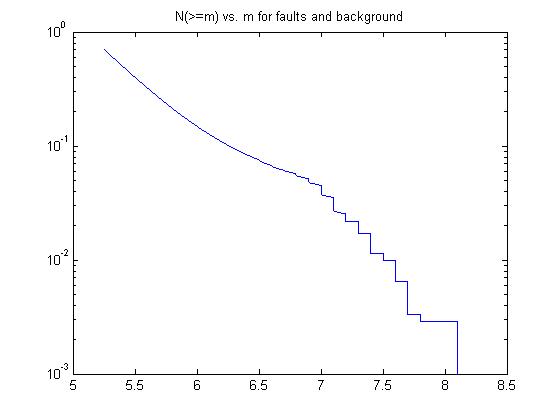
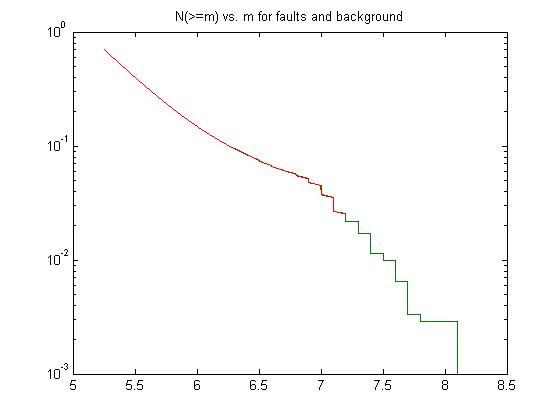
 

Fig. 5a. Num. earthquakes ≥ m vs. magnitude m Fig. 5b. Num. earthquakes ≥ m vs. mag. m

(source types separate)

**4. Importance sampling results**

Importance sampling results resulted in 73,858 earthquake scenarios (100 from faults; 73,758 from background sources. Each background location has 5 depths and approximately 20 magnitude bins)

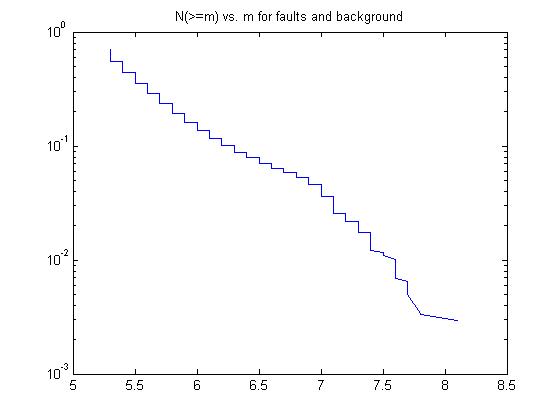
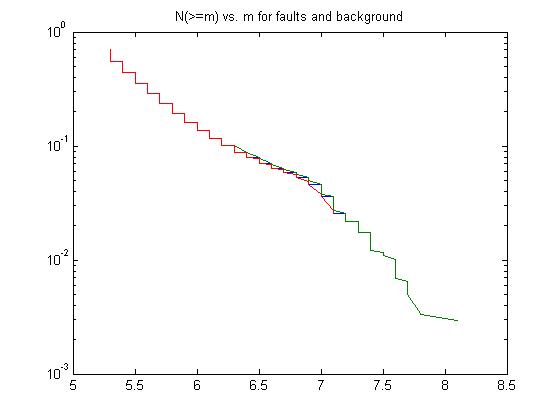
 

Fig. 6a. Num. earthquakes ≥ m vs. magnitude m Fig. 6b. Num. earthquakes ≥ m vs. mag. m

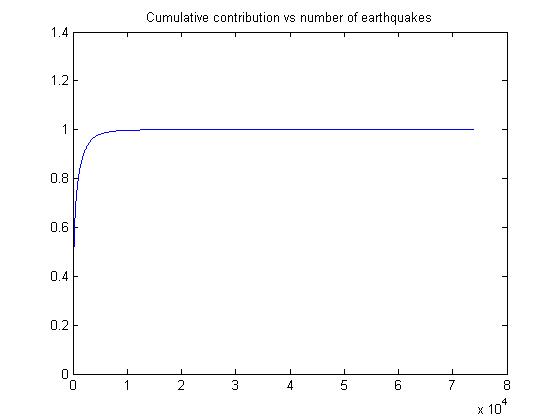
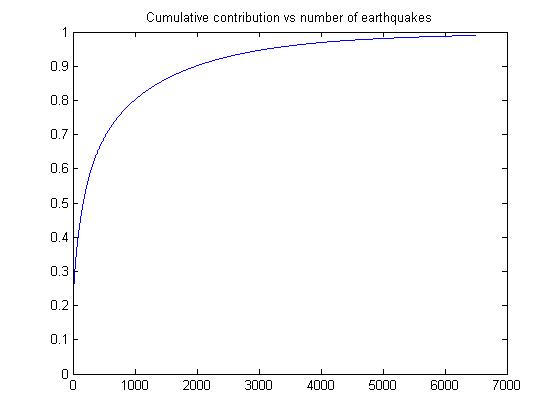
(source types separate)

**5. Reduction in importance sampling results based on contribution to hazard curves results (Step 2)**

Earthquake scenarios *j* were sorted based on their contributions *Cj* to the hazard as defined in Han and Davidson (2012):

(3)

where is annual occurrence probability of earthquake scenario *j*; is the “true” ground motion for return period *r* at site *i*; is the ground motion value at site *i* caused by earthquake *j*; and *I*, *J*, and *R* are the number of sites, earthquake scenarios, and return periods, respectively.

  Fig. 7a. Cumulative contribution vs. num. eqs Fig. 7b. Cumulative contribution vs. num. eqs

included for all 73,858 eqs from IS included, up to 99%

Table 2. Number of earthquake scenarios selected for each cumulative contribution cutoff

|  |  |  |
| --- | --- | --- |
| Cumulative contribution (%) | Exact number of earthquake scenarios included | Number of earthquake scenarios included for sensitivity |
| 99 | 6491 | 6500 |
| 97 | 4059 | 4000 |
| 95 | 3131 | 3000 |
| 90 | 1987 | 2000 |
| 85 | 1382 | 1500 |
| 80 | 995 | — |

**6. Reducing earthquake scenarios using optimization (Step 3)**

We ran the first implementation of the optimization (to reduce the number of earthquake scenarios) for 25 combinations of numbers of earthquake scenarios in the candidate, *Njc*, and reduced set, *Njr*. In each case, the candidate set comprised the *Njc* earthquake scenarios with the largest contributions. Comparison of the objective function values and MHCEs across the runs show that while there is some improvement as the *Njc* increases, there is none as the *Njr* does. While we originally expected the objective function to decrease (improve) as *Njr* increased, the reason it did not is that in each case, only about 100 earthquake scenarios are actually required to get an excellent match (Tables 3, 4, and 5).

Based on these results, in the next step, we continued with only the results from the 12 runs with *Njc*=6500 or *Njc*=4000 earthquake scenarios in the candidate set.

Table 3. Objective function value



Table 4. Mean hazard curve errors (MHCEs)



Table 5. Number of scenarios actually selected for reduced set, Njr(actual)



1. **Generating candidate ground motion maps (Step 4)**

Based on the results from the first optimization (Section 5), in the next step, we continued with only the results from the 12 runs with *Njc*=6500 or *Njc*=4000 earthquake scenarios in the candidate set. For each of those 12 runs, we generated 7 sets of candidate ground motion maps of size *Nmc*=100, 500, 1000, 2000, 3000, 4000, and 5000 (when *Njr(actual)*>*Nmc*, the run was omitted). This resulted in 77 runs. In each run, we generated the candidate set of ground motion maps using the method to reduce sampling variability by sampling allowing different numbers of ground motion maps for each earthquake scenario.

In general, the mean hazard curve error (*MHCE*) was reduced as the number of candidate maps, *Nmc*, increased or the number of ground motion maps per earthquake scenario increased, but there is variability given the number of candidate maps so the relationship is not clean. This is probably because the results from the optimization were so similar (Section 5). The MHCEs are on par with those from the Los Angeles case study.

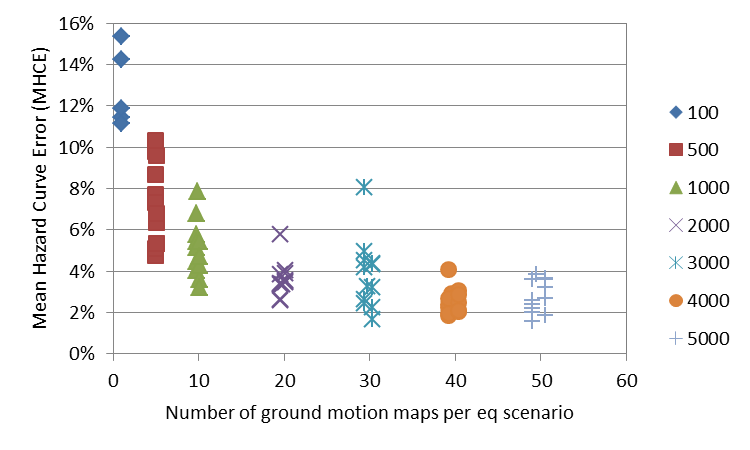


Figure 8. MHCE vs. num. maps per eq scenario

(different symbol for each *Nmc*)

Based on these results, we proceeded to the next step with only 7 runs, those with the minimum MHCE for each number of candidate maps, *Nmc*.

Table 6. Summary results from generation of candidate ground motion maps



**8. Reducing ground motion maps using optimization (Step 5)**

Based on the results from the candidate ground motion map sensitivity (Section 6), in the next step, we ran the second implementation of the optimization (to reduce the number of ground motion maps) for the last 6 of the 7 runs shown in Table 6. We did each run allowing the total number of ground motion maps in the reduced set to be *Nmr*=300 or 500. (Runs with *Nmc*=100 were not included because *Nmc*<300 already in that case.) For *Nmc*=507, 3005, or 5001, we also did runs with a full set of *Nmr*=25, 50, 75, 100, 125, and 150 as well to see how the results vary with the allowable number of ground motion maps in the reduced set (Table 7). Figures 9a and 9b show how the objective function value (i.e., weighted hazard curve errors) vary with the number of ground motion maps allowed in (Fig. 9a) and actually in (Fig. 9b) the reduced set of ground motion maps. It is apparent that the error goes to zero for many of the runs, meaning that there is no difference at all between the “true” and reduced set hazard curve values at the four specified return periods! Table 7 also shows the errors in terms of MHCE (horizontal difference between hazard curves instead of vertical, which the objective function computes). The MHCEs are computed by comparison to both the GNS-provided PSHA “true” hazard curves and the MCS “true” hazard curves. Note that in cases where the objective function is zero, the MHCE values should actually be zero as well. They are not exactly because of small interpolation error. We include the comparison to the MCS-based “true” curves to allow evaluation of spatial correlation errors.

For the comparison with the MCS-based “true” hazard curves, Table 7 shows the mean spatial correlation error (MSCE) as well. The spatial correlation pattern for the study region is represented by a matrix of weighted correlation coefficients, where the weights are the annual ground motion map occurrence probabilities, *Pn*. For each pair of sites *a* and *b*, where *yan* and *ybn* are, respectively, the ground motions at sites *a* and *b* in ground motion map *n*, the weighted correlation coefficient *ρa,b* is:

(3)

The spatial correlation error (*SCE*) for sites *a* and *b* is defined as:

(4)

where and are the weighted correlation coefficients calculated in Equation 3 using, respectively, the full set of 1,000,000 years of “true” MCS ground motion maps, or the reduced set of probabilistic ground motion maps. A summary measure of the overall spatial correlation match then, is the mean, over all pairs of sites, of the absolute value of *SCE* (Eq. 5).

(5)

Table 7. Summary results from reduction of candidate ground motion maps



As shown in Table 7, the spatial correlation errors are very small, negligible for practical use as well.

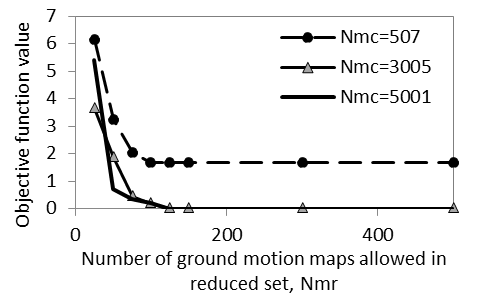
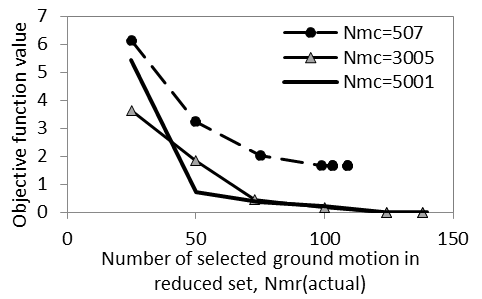
 

Fig. 9a. Objective function value vs. Num. Fig. 9b. Objective function value vs. Num.

ground motion maps allowed in reduced set, *Nmr* GM maps actually in reduced set, *Nmr(actual)*

**9. Conclusions**

**9.1. Final recommended reduced set of ground motion maps**

There are many promising and comparable sets of results (i.e., any of the ones with objective function of zero). As an example, for Run 11 (2003 candidate ground motion maps, 300 allowed in reduced set, 138 actually selected in reduced set), we have the histograms of HCEs and SCEs in Figures 10 and 11. The errors are negligible when compared to the “true” GNS-based hazard curves used to develop the reduced set. When compared to the MCS-based “true” curves they are a bit larger just because of the difference in those “true” curves. The distribution of spatial correlation errors (Fig. 11) shows small errors, a bit skewed due to the difference in the GNS-based and MCS-based “true” hazard curves. If the process was conducted all the way through using the MCS-based “true” hazard curves, that skew should go away. Figure 12 shows the fault and background sources selected for this run. Some sources include multiple events, so there are fewer than 138 sources shown. The solution includes 110 ground motion maps generated from background sources and 28 from fault sources. The events range in magnitude from 5.26 to 7.24, and the adjusted annual occurrence probabilities from 4.48(10-7) to ­7.40(10-4). Finally, Figures 13 and 14 show comparisons between the full reduced set hazard curve for the site with the worst HCEs, plotted together with the “true” hazard points being matched (Fig. 13) and with the full hazard curve from the one million year MCS run. Again, the match is great at the specified points. It is not as good at points that are not the ones specified to match to, especially on the leftmost and rightmost parts of the curves. If desired, one could certainly rerun the analysis but specifying more return periods to match to and that would remedy the situation.

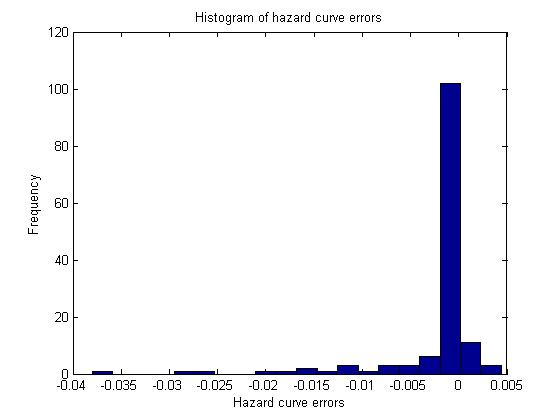
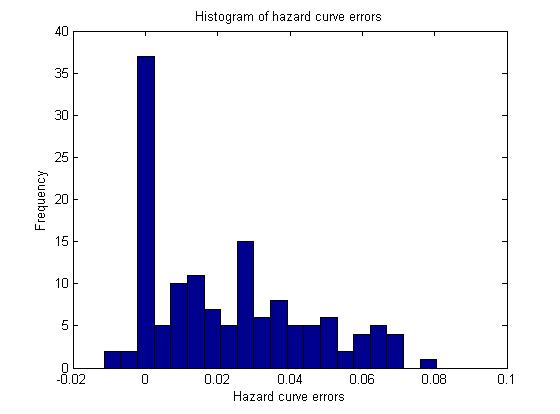
 

Fig. 10a. Histogram of hazard curve errors Fig. 10a. Histogram of hazard curve errors

based on comparison with GNS “true” curves based on comparison with MCS “true” curves

for Run 11 for Run 11

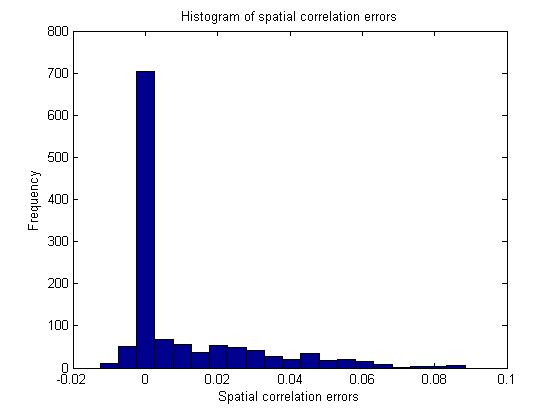
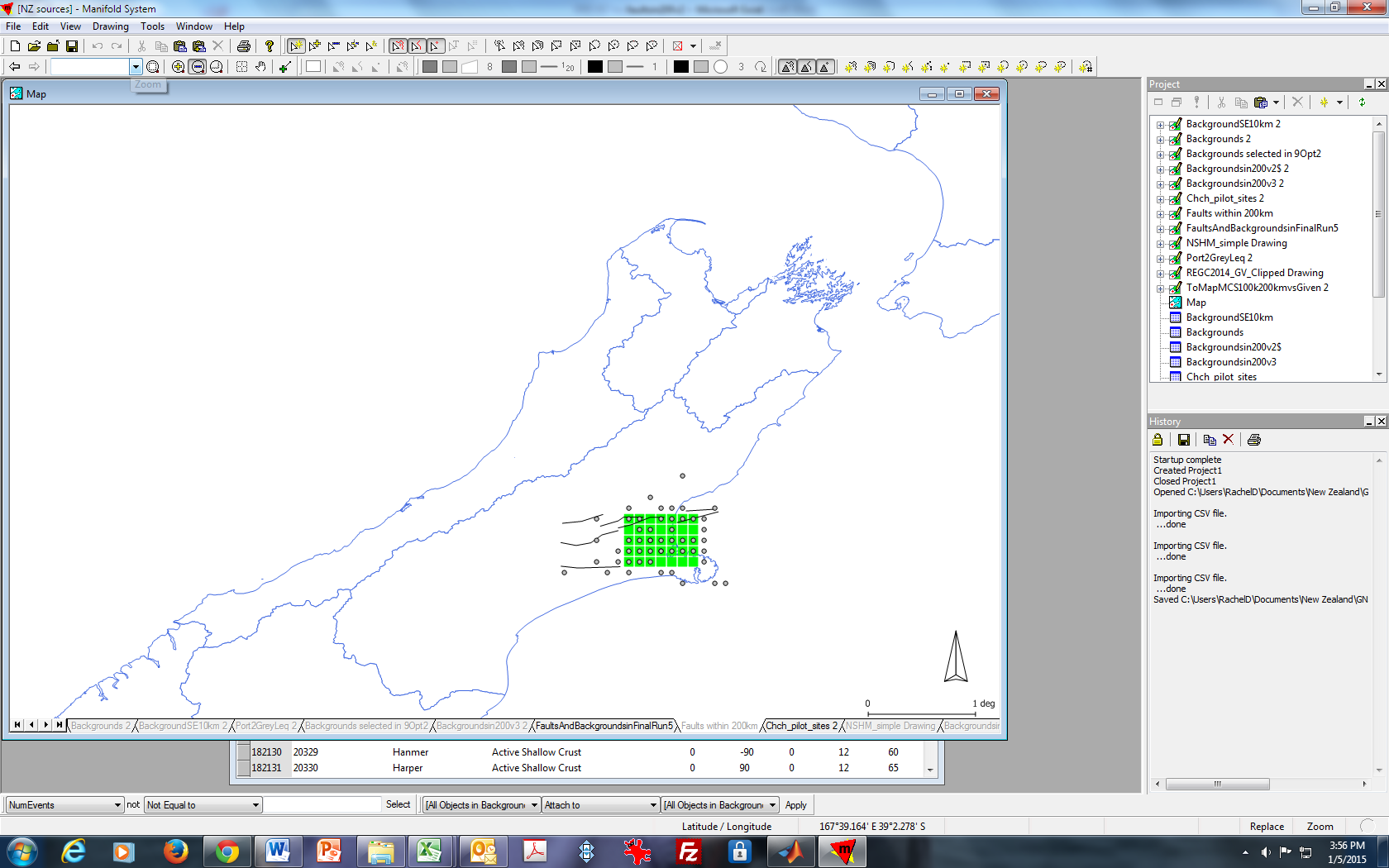
**** 

Fig. 11. Histogram of spatial correlation errors Fig. 12. Map of selected background and fault sources

for Run 11 in Run 11

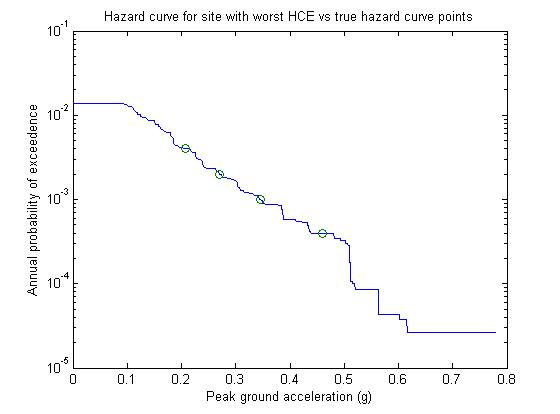
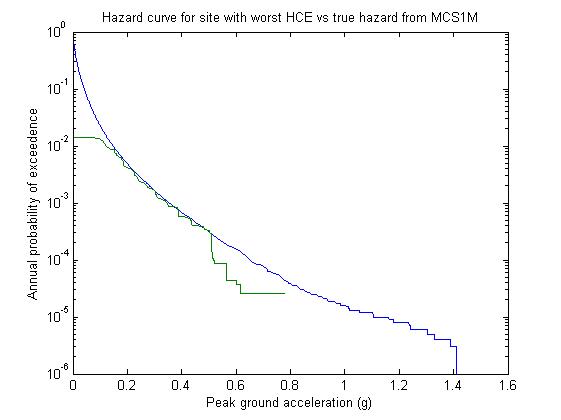
 

Fig. 13. Run 11 hazard curve vs “True” Fig. 14. Run 11 hazard curve vs hazard curve

points matched to, for site with worst HCEs from MCS1M, for site with worst HCEs

As another example, for Run 28 (5001 candidate ground motion maps, 150 allowed in reduced set, 123 actually selected in reduced set), we have the histograms of HCEs and SCEs in Figures 15 and 16. The solution includes 94 ground motion maps generated from background sources and 29 from fault sources. The events range in magnitude from 5.25 to 7.24, and the adjusted annual occurrence probabilities from 1.06(10-5) to 6.79(10-4). This solution basically removed the events with the lowest probabilities without making the match worse. This solution is equally good in that the objective function value is also zero and the spatial correlation errors are similarly negligible. The hazard curve errors shown in Figure 15a are due to interpolation and those in Figure 15b are also due to the difference between the GNS PSHA results and the UD MCS results. This result from Run 28 thus may be preferred because it includes only 123 ground motion maps (compared to the 138 for Run 11).

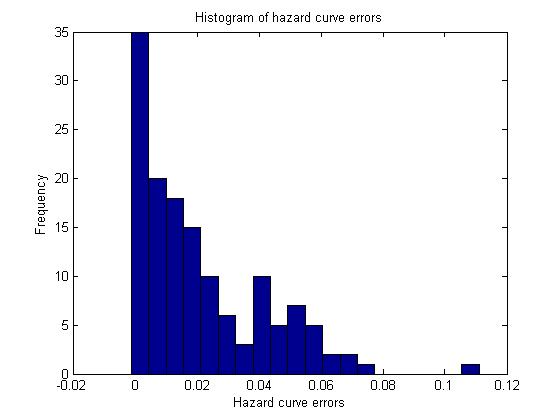
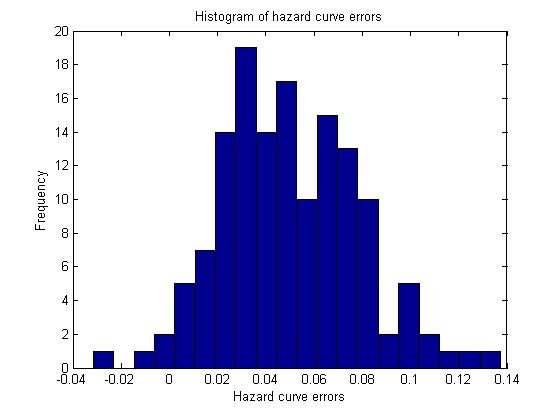
** **

Fig. 15a. Histogram of hazard curve errors Fig. 15a. Histogram of hazard curve errors

based on comparison with GNS “true” curves based on comparison with MCS “true” curves

for Run 28 for Run 28

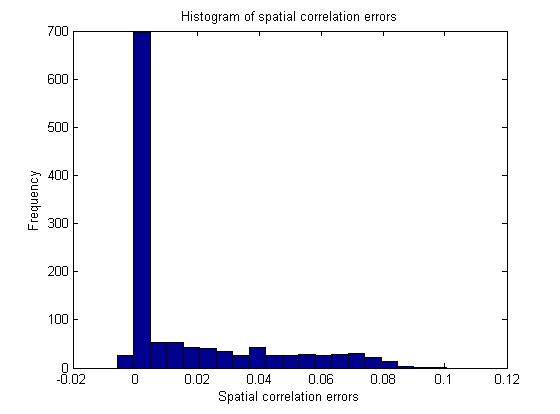
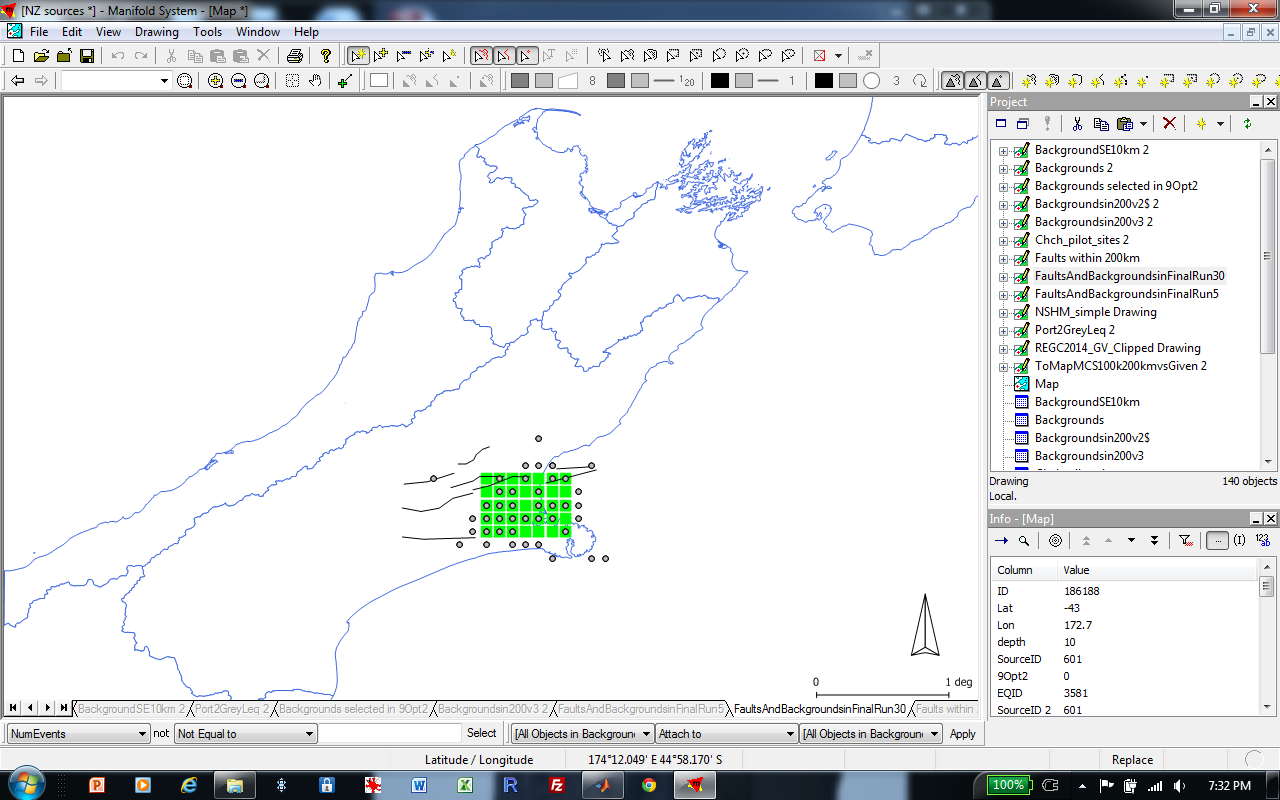
**** 

Fig. 16. Histogram of spatial correlation errors Fig. 17. Map of selected background and fault sources

for Run 28 in Run 28

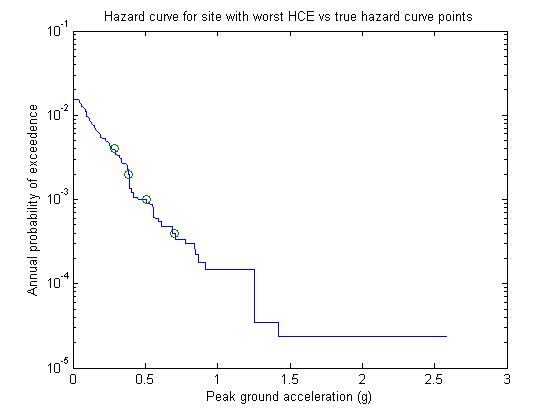
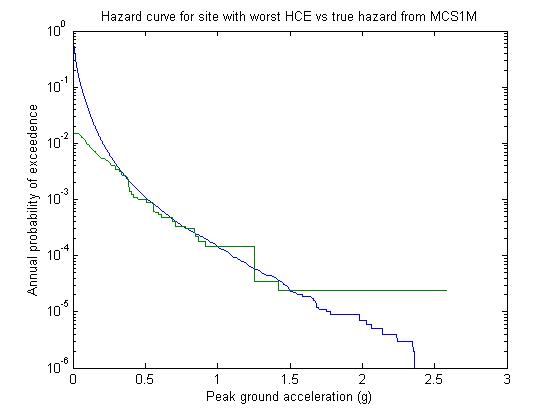
 

Fig. 18. Run 28 hazard curve vs “True” Fig. 19. Run 28 hazard curve vs hazard curve

points matched to, for site with worst HCEs from MCS1M, for site with worst HCEs

**9.2. Conclusions and recommendations for applying method**

**Some general conclusions**

* The three methods of estimating hazard—PSHA, Monte Carlo Simulation (MCS), and MCS with importance sampling of magnitudes—provide comparable results (Section 2).
* Monte Carlo Simulation requires is extremely computationally demanding (Section 3). Because of the high seismicity in New Zealand (many background sources and high recurrence rates), the simulation must be run on the order of a million years, generating hundreds of thousands of earthquake scenarios.
* Even when importance sampling of the magnitudes is used to make MCS more efficient, the process is still very computationally intensive due to the number of background sources (Section 4). With importance sampling the process still generated about 75,000 earthquake scenarios, making any subsequent analysis very computationally intensive.
* A very small percentage of the earthquake scenarios contribute the vast majority of the hazard curve information, allowing another opportunity for efficiency (Section 5). In this Christchurch example, approximately 6500 (9%) of the 75,000 earthquake scenarios generated with importance sampling provide 99% of the contribution to the hazard curves.
* The proposed OPS-based method provides excellent results, ultimately offering no error in the hazard curves (at the specified sites and return periods) and negligible error in the spatial correlation, with just about 140 ground motion maps.
* There are multiple equally good solutions (with zero hazard curve error).

**Guidance in implementing the method**

* In applying the overall method, one has to set a key parameter at each step (contribution threshold in Step 2, *Jred* in Step 3, total number of candidate ground motion maps in Step 4, *Jred* in Step 5) and it is not clear what combination of those provides the best optimization over the whole process. For example, should one generate more earthquake scenarios and fewer ground motion maps per earthquake scenario? Or fewer earthquake scenarios and more ground motion maps per earthquake scenario? Following are some guidelines to help determine how to set these parameters in each step.
* Use importance sampling of magnitudes because it is theoretically well-grounded and makes the initial sampling of candidate earthquake scenarios more efficient without any loss in final results.
* Do an initial reduction in the set of earthquake scenarios based on contribution to the hazard curves. The exact percentage used as a cutoff for cumulative contribution may depend on the application, but one should try to set it so as to retain on the order of thousands of earthquake scenarios and thus to have sufficient candidate earthquakes to choose from in the optimization. The first implementation of the optimization is easy to solve, so having more candidate earthquake scenarios (say 6500 instead of 1500) does not affect the ability to solve it, and thus it is better to err on the side of more. Nevertheless, there is no value in including more than those providing 99% of the contribution. The cutoff is not likely to be crucial; the point is just to retain enough candidate earthquake scenarios but not so many extra that the optimization becomes more cumbersome or difficult to solve without adding any benefit. For a high seismicity area like New Zealand, about 99% seemed appropriate. This step is very simple and quick to do and offers enormous savings in focusing on the most promising earthquake scenarios and thus making the first implementation of the optimization easier to solve.
* In the first implementation of the optimization (reducing the set of earthquake scenarios), if the initial reduction has been done, the optimization seems to require only very few scenarios in the reduced set, so one should start by allowing a larger number in the reduced set (say 1000) and seeing how many the optimization chooses. If it uses all allowed (i.e., 1000 in that case), one can try increasing the *Jred* value until it does not. It is likely that one can just use the maximum number that the optimization wants to select (e.g., about 100 in this Christchurch example).
* In deciding how many candidate ground motion maps to generate, having too few will result in larger errors, but at a certain point there is no benefit to having more. For the Christchurch case study, about 3000 candidate ground motion maps was sufficient and more did not provide a reduction in the errors. The second implementation of the optimization is much more difficult to solve than the first, so one does not want to have more than several thousand candidate ground motion maps or it will be harder for the optimization to find a good solution. There is a tradeoff between having enough candidate ground motion maps so there are enough choices for the optimization to find a good solution, but not so many that it becomes too computationally intensive to solve.
* In general, one can set the parameters (contribution threshold in Step 2, *Jred* in Step 3, number candidate maps in Step 4, *Jred* in Step 5) as one thinks is reasonable following these guidelines, then see if the resulting errors are acceptable at each stage, and modify them accordingly if necessary.

**APPENDIX D.**

**Gurobi vs COIN comparison**

Although Gurobi is a commercial solver that is known to perform very well, it is not free for GNS, and thus we compared its performance to that of the free solver, PULP\_CBC\_CMD, from COIN-OR. We conducted several runs of the second implementation of the optimization, allowing both solvers to run for a maximum of 5 hours in each case (Table 1). (Both solvers can find the optimal solution for the first implementation of the optimization without a problem.) The results suggest that COIN works as well as Gurobi when the number of maps allowed in the reduced set (*Nmr*, or *Jred* in the optimization) is 100 or larger. For the cases when the number of maps allowed in the reduced set is smaller, Gurobi performs better, although it may be that COIN would do better if allowed to run for longer.

This brief comparison suggests that COIN should be adequate for most practical cases, although it may be slower than Gurobi. If there is concern that the COIN solution is not sufficient, the user could let it run longer (say overnight) to find an improved solution. If that does not work, or for larger problems, one can try switching to Gurobi, CPLEX, or another commercial solver.

Table 1. Comparison of performance with Gurobi and COIN solvers



1. Note that if the analysis is changed so that ruptures of different lengths are possible for a single source, the Rrup would have to be calculated separately for each earthquake scenario just before *GroundMotionNZ.m* is called in *SeisEventNZ.m*. [↑](#footnote-ref-1)