OPENQUAKE ENGINE

RISK QA REPORT

Version 1.0.0

Testing procedures and quality assurance methods adopted in the development of the risk component of the OpenQuake Engine, an open source code for seismic hazard and physical risk calculation.

GEM
GLOBAL EARTHQUAKE MODEL
working together to assess risk

“OpenQuake: Calculate, share, explore”

Testing procedures adopted in the development of the risk component of the OpenQuake-engine

Authors
Anirudh Rao[1], Michele Simionato[1]

[1] GEM Model Facility
via Ferrata, 1
20133 Pavia
Italy

Email address (for all the authors):
<name.surname>@globalquakemodel.org

Citation
Please cite this document as:
Rao, Anirudh and Simionato, Michele (2015). Testing procedures adopted in the development of the risk component of the OpenQuake-engine. *Global Earthquake Model (GEM) Technical Report 2015-08. doi: 10.13117/GEM.OPENQUAKE.TR2015.08, 90 pages.*

First printing, August 2015

Contents

Part I

Introduction

1. Software Testing

The current document describes the testing procedures adopted in the development of the hazard component of the OpenQuake-engine (OQ-engine), the open source hazard and risk software developed by the Global Earthquake Model initiative.

Nowadays seismic hazard analysis serves different needs coming from a variety of users and applications.

These may encompass engineering design, assessment of earthquake risk to portfolios of assets within the insurance and reinsurance sectors, engineering seismological research, and effective mitigation via public policy in the form of urban zoning and building design code formulation.

Decisions based on seismic risk results may have impacts on population, properties and capitals, possibly with important repercussions on our day-to-day life. For these reasons, it is recommendable that the generation of hazard models and their calculation is based on well-recognized, state-of-the-art and tested techniques, requirements that must be reconciled with the need to regularly incorporate recent advances given the progress carried out within the scientific community.

The features described below contribute to fulfill these requirements:

- Software should have a modular and flexible structure capable of incorporating new features and - as a consequence - offer users the most recent and advanced techniques. In very general terms, modularity is the level to which a component of a system can be moved, replaced or reused. In software design, modularity means the separation of the software into smaller independent components that can be implemented, maintained and tested easily and efficiently.
- Software should have and extensive test coverage which captures possible errors and avoids regressions (i.e. unexpected behaviors introduced by new features). Software testing (Myers et al., 2012) is an important, complex and vast discipline which helps in developing methods and processes aimed at certifying the extent to which a computer code

behaves according to the original design intent and user specifications.

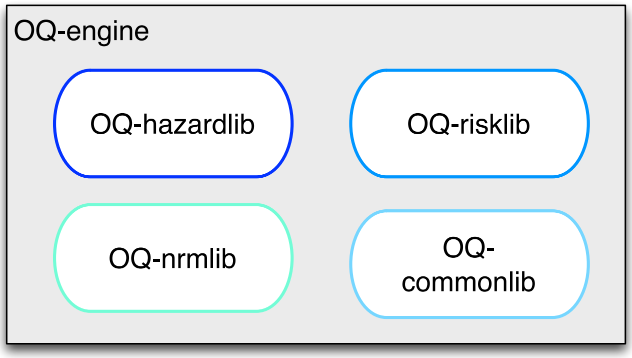

Figure 1.1 – *A schematic describing the main components of the OpenQuake-engine software.*

The OQ-engine includes different levels of modularity. The first is the one separating the engine itself into a number of libraries (see Figure 1.1), each one containing well identified knowledge, objects and methods (e.g. the OQ-hazardlib includes objects and methods needed to compute probabilistic seismic hazard and the OQ-risklib contains methods to compute scenario and probabilistic seismic risk).

The second one pertains to the data model adopted in the development of each library as a result of the abstraction process.

According to Berkes (2012) scientific software must be:

- Error proof
- Flexible and able to accommodate different methods
- Reproducible and re-usable

1.1 Testing and Quality Assurance

Despite the distinction between software testing (in some cases also called Quality Control) and Software Quality Assurance (SQA) being somewhat vague and partly open to personal judgment, it's clear that SQA is a more comprehensive and overarching process than software testing. SQA aims at the definition of the best processes that should be used to provide guarantees that user expectations will be met. Software testing focuses instead on detecting software faults by inspecting and testing the product at different stages of development.

1.1.1 Software testing

Software testing can be implemented at different stages of the development process, with varying strategies to approach the problem. The OQ-engine and the associated libraries are developed following an agile paradigm. This development strategy is organized in a way that the creation of the real code is completed in parallel and fully integrated with the software testing process.

The software engineering community provides a wide range of testing levels and typologies. In the current document we consider just a portion of them with the specific intent to illustrate the standards used in the development of the OQ-engine and particularly of its risk calculation component.

1.1.2 Quality assurance

From the IEEE "Standard for Software Quality Assurance Processes": *Software quality assurance is a set of activities that define and assess the adequacy of software processes to provide evidence that establishes confidence that the software processes are appropriate for and produce software products of suitable quality for their intended purposes. A key attribute of SQA is the objectivity of the SQA function with respect to the project. The SQA function may also be organizationally independent of the project; that is, free from technical, managerial, and financial pressures from the project.* In this document we are not covering topics related to SQA since this would go beyond its scope.

1.2 Organization of Report

This document is organized into four chapters.

The current chapter provides a very brief and general introduction to software testing with a focus on the testing of scientific software.

The second chapter describes the module, or unit testing procedures adopted in the development of the OQ-engine and we discuss some examples. The continuous integration mechanism used for development is also discussed.

The third chapter describes the general framework for the acceptance tests for the OpenQuake risk calculators. A brief overview of the theoretical background for the different calculators is also provided in this chapter.

The fourth chapter describes the different test cases, input models, and results for the acceptance tests implemented for the OpenQuake scenario risk, classical risk, and event-based risk calculators.

Part II

Unit Tests

2. Unit Testing in the OpenQuake-engine

This chapter provides an introduction to the module (unit) testing procedures (Myers et al., 2012) and describes the estensive series of tests implemented in the OQ-engine.

2.1 Overview of Unit-Testing

At the first level of the code testing process is the practice of "unit-testing". This process is a central tenet of test-driven software development and is widely established as a means of "best-practice". Before looking closely at the OpenQuake-engine approach to unit-testing it is important to establish what are the precise objectives of the unit-testing process and the benefits (and limitations) that it brings.

2.1.1 Correctness of implementation

This objective is obviously the primary goal of unit-testing, to ensure that each function of the code is operating in the manner expected by the developer. "Correctness", in this case, requires that the function produces both the correct output, but also if there are cases in which function may fail then the means of failure should be predictable. The following is a relatively simple example of how a unit-test relates to a function:

Consider a simple function to multiply two numbers and take the logarithm of the result. A relevant analogy may be that of a magnitude scaling relation calculation, in which both a rupture length and rupture width are required, and the logarithm of the area may be needed by the function itself. In this circumstance a negative value in either of the two inputs would result in a calculation error. This could be coded in the following manner:

```
def get_log_area(length, width):
    if (length < 0) or (width < 0):
        raise valueError("Both inputs must be positive")
    else:
```

```
        return log10(length * width)
```

From the description above it is evident that the user requirements inform the manner in which the function should behave (i.e. negative values cannot be tolerated). To ensure that the function is operating correctly, we wish to write a set of tests that will confirm the behaviour is correct:

1. If both a and b are equal to 10.0, then the function should return 2.0
2. If $a = -1$ and $b = 10$ the function should raise an error reporting the stated message "Both inputs must be positive".
3. If $a = 10$ and $b = -1$ the function should raise an error reporting the stated message "Both inputs must be positive".
4. If $a = -1$ and $b = -1$ the function should raise an error reporting the stated message "Both inputs must be positive".

A unit-test for this function is an additional function that will check that both cases are satisfied, and will report an error if not. A comprehensive unit-test suite for a software may fulfil two objectives: **line coverage** and **parameter coverage**. The former should ensure that, in as far as possible, every line (or statement) in the code is executed at some point in the testing process. The latter should ensure that the behaviour of the function is predictable when supplied with "unusual" parameters. In the above example, both objectives are satisfied by the tests. The first test will result in a positive valued "area", thus executing the second branch of the logical path, the second test will result in a negative area and will execute the first logical branch.

Therefore all lines of the code are covered and the line coverage is complete. We also see that in this simple example there are four possible cases: i) a is positive and b is positive, ii) a is positive and b is negative, ii) a is negative and b is positive, and iv) both a and b are negative. Only the first case is valid, therefore the first test ensures that they provide the correct answer (usually verified by independent means), whilst the remaining tests should ensure that the function raises the correct error. Thus the full parameter space of the input is ensured.

The above case is, of course, trivial; however, as shall be seen in due course, this same process can be applied in more complex contexts.

Furthermore, the same unit-testing approach can be applied not only to individual components within the hazard and risk calculations, but also to full calculations, essentially verifying that the hazard curves and loss curves produced by the full OpenQuake probabilistic hazard and risk calculators are in agreement with those produced independently (sometimes by hand calculations).

2.1.2 Identify problems prior to software release

This advantage is largely self-explanatory, but for many software projects this can reduce the possibility of requiring *a posteriori* fixes to the code (patches). By compiling a comprehensive suite of unit-tests, and following a software development and release process that should automatically run the tests at the point of packaging, this should ensure that new features added to the software cannot inadvertently break other components.

2.1.3 Facilitate improvements in performance

In the creation of software intended to perform demanding scientific calculations, like those commonly associated with probabilistic seismic hazard and risk analysis, the issue of computational performance and efficiency is a major one. There is a continuing need to improve the speed and reduce the work required to undertake the haard and risk calculations. To implement improvements it is necessary to ensure that optimisations do not modify the outputs of the calculation, only the speed at which they are performed. Thus, unit-testing is absolutely fundamental to this process as optimisation cannot be undertaken readily without a means to ensure the calculation outputs have not changed.

2.2 Continuous Integration

OQ-engine is developed and packaged within a "continuous integration" system (https://ci.openquake.org/), based on the open-source software "Jenkins" (http://jenkins-ci.org/). Continuous integration is used in large software projects to run a full test suite of the complete software, either at fixed time intervals or, as in the current case, when any new code is committed to the repository. The continuous integration system does the following:

1. Run the full set of unit-tests for all code in all of the linked repositories. This will include the main (or "master") branch of the software repository, i.e. the one that will be used for packaging of the software, as well as some development branches.
2. Run a test of the software installation. This test will install the software on a dedicated platform and check that the installation of the software is successful. This test also ensures that if changes occur in the dependency packages, and these changes affect or compromise the installation and operation of the software, these problems are recognised immediately.
3. The software will also run standard Python tests for quality of code, compilation of documentation etc.
4. Several long-running tests may also be run. These implement larger scale seismic hazard and risk calculations designed to test the overall performance of the engine.

If at any point the tests should fail, the OpenQuake development team will be notified automatically. This ensures that software that is failing any of the tests will remain on the main branch of the repository for the minimum amount of time possible. Furthermore, if the continuous integration tests fail, the new code will not be integrated into the nightly package of the software.

2.3 Unit-Tests in the OpenQuake Risk Library

2.3.1 Component level tests

The unit-testing at the component level breaks the functions into simple calculations whose results can be verified by hand. These tests, similar in nature to that illustrated previously, provide the majority of the line and parameter coverage needed to ensure a robust code.

To illustrate the comprehensive nature of the coverage we consider the example of the lognormal distribution function used for representing continuous vulnerability functions:

The test suite for this one function is illustrative of several key components of the unit-testing. First is the use of an independent tool to provide the expected values of the calculation under simple conditions. Second is the use of "extreme cases" such as polar locations, or across the International Dateline. These ensure that the function can be global in application.

The nature of the interdependencies between the functions also means that one a functions own unit-test is verified, the function can then form the basis for testing other conditions. So for example, the geodetic distance tools also contain a method to calculate the minimum distance between a collection of points and a single point. Rather than requiring new expected distances for the different conditions, the geodetic distance function can then be used to construct tests for functions that utilise it. This makes the testing process more efficient, and reduces the need to write large numbers of tests in order to ensure correct behaviour of the function.

2.4 Summary

In this chapter we have outlined both the process and the key benefits of developing comprehensive unit-tests for OpenQuake-engine, as well as outlining the operation of the continuous integration system, which should ensure that code with the potential to break the tests cannot be packaged and released. The unit-tests themselves have not been discussed in detail as nearly one thousand tests are executed during the unit-test process. However, to view the comprehensive set of tests, the reader is encouraged to refer to the full test-suite, which is open and available on the OpenQuake code repository (https://github.com/gem/oq-risklib/tree/master/openquake/risklib/tests).

Part III

Acceptance Tests

3. Framework for Acceptance Testing

3.1 Verification Framework

The main purpose of the acceptance tests is to ensure that the risk calculators work according to the design specifications and to verify that the calculators produce correct results for a variety of input cases. Correctness of the test case results is verified by comparing with hand calculations for the simple test cases or with alternate implementaions in Julia for the complex cases.

3.2 Theoretical Background

3.2.1 Scenario risk

The scenario risk calculator computes loss statistics for all assets in a given exposure model for a single specified earthquake rupture. Loss statistics include the mean and standard deviation of ground-up losses and insured losses for each loss type considered in the analysis. Loss statistics can currently be computed for five different loss types using this calculator: structural losses, nonstructural losses, contents losses, downtime losses, and occupant fatalities. This calculator requires the definition of a finite rupture model, an exposure model and a vulnerability model for each loss type considered; the main results are the loss statistics per asset and mean loss maps.

The rupture characteristics—i.e. the magnitude, hypocenter and fault geometry—are modelled as deterministic in the scenario calculators. Multiple realizations of different possible ground motion fields (GMFs) due to the single rupture are generated, taking into consideration both the inter-event variability of ground motions, and the intra-event residuals obtained from a spatial correlation model for ground motion residuals. The use of logic-trees allows for the consideration of uncertainty in the choice of a GMPE model for the given tectonic region and in the choice of vulnerability functions for the different taxonomy types in the exposure model.

As an alternative to computing the GMFs with OpenQuake, users can also provide their own sets of GMFs as input to the scenario risk calculator.

For each GMF realization, a loss ratio is sampled for every asset in the exposure model using the provided probabilistic vulnerability model, taking into consideration the correlation model for vulnerability of different assets of a given taxonomy. Finally loss statistics, i.e., the mean loss and standard deviation of loss for both ground-up losses and insured losses across all realizations, are calculated for each asset. Mean loss maps are also generated by this calculator, describing the mean ground-up losses and mean insured losses caused by the scenario event for the different assets in the exposure model.

3.2.2 Scenario damage

The scenario damage calculator computes damage distribution statistics for all assets in a given exposure model for a single specified earthquake rupture. Damage distribution statistics include the mean and standard deviation of damage fractions for different damage states. This calculator requires the definition of a finite rupture model, an exposure model and a fragility model; the main results are the damage distribution statistics per asset, aggregated damage distribution statistics per taxonomy, aggregated damage distribution statistics for the region, and collapse maps.

The rupture characteristics—i.e. the magnitude, hypocenter and fault geometry—are modelled as deterministic in the scenario calculators. Multiple realizations of different possible ground motion fields (GMFs) due to the single rupture are generated, taking into consideration both the inter-event variability of ground motions, and the intra-event residuals obtained from a spatial correlation model for ground motion residuals. The use of logic-trees allows for the consideration of uncertainty in the choice of a GMPE model for the given tectonic region and in the choice of fragility functions for the different taxonomy types in the exposure model.

As an alternative to computing the GMFs with OpenQuake, users can also provide their own sets of GMFs as input to the scenario damage calculator.

For each GMF realization, damage fractions (the fraction of buildings in each damage state) are estimated for every asset in the exposure model using the provided fragility model, and finally the damage distribution statistics (i.e., the mean damage fractions and standard deviation of damage fractions for all damage states) across all realizations are calculated. The calculator also provides aggregated damage distribution statistics for the portfolio, such as mean damage fractions and standard deviation of damage fractions for each taxonomy in the exposure model, and the mean damage fractions and standard deviation of damage fractions for the entire region of study.

3.2.3 Classical PSHA-based risk

The classical PSHA-based risk calculator convolves through numerical integration, the probabilistic vulnerability functions for an asset with the seismic hazard curve at the location of the asset, to give the loss distribution for the asset within a specified time period. The calculator requires the definition of an exposure model, a vulnerability model for each loss type of interest with vulnerability functions for each taxonomy represented in the exposure model, and hazard curves calculated in the region of interest. Loss curves and loss maps can currently be calculated for five different loss types using this calculator: structural losses, nonstructural losses, contents losses, downtime losses, and occupant fatalities. The main results of this calculator are loss exceedance curves for each asset, which describe the probability of exceedance of different loss levels over the specified time period, and loss maps for the region, which describe the loss values that have a given probability of exceedance over the specified time period.

The hazard curves required for this calculator can be calculated by the OpenQuake engine for all asset locations in the exposure model using the classical PSHA approach (Cornell, 1968; McGuire, 1976). The use of logic-trees allows for the consideration of uncertainty in the choice of a GMPE model for the different tectonic region types in the region and in the choice of vulnerability functions for the different taxonomy types in the exposure model.

3.2.4 Classical PSHA-based damage

The classical PSHA-based risk calculator convolves through numerical integration, the probabilistic vulnerability functions for an asset with the seismic hazard curve at the location of the asset, to give the loss distribution for the asset within a specified time period. The calculator requires the definition of an exposure model, a vulnerability model for each loss type of interest with vulnerability functions for each taxonomy represented in the exposure model, and hazard curves calculated in the region of interest. Loss curves and loss maps can currently be calculated for five different loss types using this calculator: structural losses, nonstructural losses, contents losses, downtime losses, and occupant fatalities. The main results of this calculator are loss exceedance curves for each asset, which describe the probability of exceedance of different loss levels over the specified time period, and loss maps for the region, which describe the loss values that have a given probability of exceedance over the specified time period.

The hazard curves required for this calculator can be calculated by the OpenQuake engine for all asset locations in the exposure model using the classical PSHA approach (Cornell, 1968; McGuire, 1976). The use of logic-trees allows for the consideration of uncertainty in the choice of a GMPE model for the different tectonic region types in the region and in the choice of vulnerability functions for the different taxonomy types in the exposure model.

3.2.5 Event-based risk

This calculator employs an event-based Monte Carlo simulation approach to probabilistic risk assessment in order to estimate the loss distribution for individual assets and aggregated loss distribution for a spatially distributed portfolio of assets within a specified time period. The calculator requires the definition of an exposure model, a vulnerability model for each loss type of interest with vulnerability functions for each taxonomy represented in the exposure model, and a set of ground motion fields representative of the seismicity of the region over the specified time period. Loss curves and loss maps can currently be calculated for five different loss types using this calculator: structural losses, nonstructural losses, contents losses, downtime losses, and occupant fatalities. The main results of this calculator are loss exceedance curves for each asset, which describe the probability of exceedance of different loss levels over the specified time period, and loss maps for the region, which describe the loss values that have a given probability of exceedance over the specified time period. Aggregate loss exceedance curves can be also be produced using this calculator; these describe the probability of exceedance of different loss levels for all assets of a single taxonomy, or for all assets in the portfolio, over the specified time period. Finally, event loss tables can be produced using this calculator; these tables describe the total loss across the portfolio for each seismic event in the stochastic event set.

This calculator relies on the probabilistic event-based hazard calculator, which simulates the seismicity of the chosen time period T by producing a *stochastic event set* (also known as a *synthetic catalog*). For each rupture generated by a source, the number of occurrences in the given time span T is simulated by sampling the corresponding probability distribution as given by $P_{rup}(k|T)$. A stochastic event set is therefore a *sample* of the full population of ruptures as defined by a seismic source model. Each rupture is present zero, one or more times, depending on its probability. Symbolically, we can define a stochastic event set (*SES*) as:

$$SES(T) = \{k \times rup,\ k \sim P_{rup}(k|T)\ \ \forall\, rup\ in\ Src\ \forall\ Src\ in\ SSM\} \tag{3.1}$$

where k, the number of occurrences, is a random sample of $P_{rup}(k|T)$, and $k \times rup$ means that rupture rup is repeated k times in the stochastic event set.

For each event in the stochastic event sets, a spatially correlated ground motion field (GMF) realisation is generated, taking into consideration both the inter-event variability of ground motions, and the intra-event residuals obtained from a spatial correlation model for ground motion residuals. The use of logic-trees allows for the consideration of uncertainty in the choice of a seismic source model, in the choice of GMPE models for the different tectonic regions, and in the choice of vulnerability functions for the different taxonomy types in the exposure model.

For each GMF realization, a loss ratio is sampled for every asset in the exposure model using the provided probabilistic vulnerability model, taking into consideration the correlation model for vulnerability of different assets of a given taxonomy. Finally loss exceedance curves are computed for both ground-up losses and insured losses.

4. Test Cases and Results

4.1 Scenario Risk Calculator

The tests for the scenario risk calculator assume the correct computation of the ground motion fields at the locations of the assets in the exposure model. Thus, the risk tests implicitly rely on the acceptance tests for the scenario hazard calculator.

The rupture model used for the tests comprises a magnitude $M_W 6.7$ rupture on a vertical strike-slip fault.

Details of the rupture are given below:

Fault type: Strike slip
Fault dip: 90°
Fault plane depths: 0–20 km
Fault coordinates:
South end: $38.0000°N$, $122.0000°W$
North end: $38.2248°N$, $122.0000°W$
Rupture magnitude: 6.7
Rupture hypocenter: $38.1124°N$, $122.0000°W$
Hypocenter depth: 10 km

The complete collection of input models and job configuration files used in these test cases can be accessed here: https://github.com/gem/oq-risklib/tree/master/openquake/qa_tests_data/scenario_risk

4.1.1 Single asset tests

The single asset test cases are designed to test the basic elements of the scenario risk calculator, such as:

- basic loss field computation

Site	Taxonomy	Latitude	Longitude	Comment
1	tax1	38.113	-122.000	On fault midpoint, along strike

Table 4.1 – *Asset location and taxonomy for the single-asset test cases*

- calculation of mean and standard deviation of scenario loss

The location and taxonomy of the single asset in the exposure model used for the single-asset test cases for the scenario risk calculator are given in Table 4.1.

4.1.1.1 Case 1a

Test Case 1a uses a set of five precomputed ground motion values to test the correct interpolation of the mean loss ratios of the vulnerability function at intermediate intensity measure levels. There is no uncertainty in the vulnerability function used for this case. The coefficient of variation of the loss ratio is zero at all intensity measure levels.

GMF #	Site	PGA (g)
1	1	1.300
2	1	0.044
3	1	0.520
4	1	1.000
5	1	1.200

Table 4.2 – *Five precomputed ground motion fields at a single site*

Table 4.2 lists the five ground motion values used in this test case.

PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
Mean LR	0.01	0.04	0.10	0.20	0.33	0.50	0.67	...	0.99
CoV LR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0

Table 4.3 – *Lognormal vulnerability function with zero coefficients of variation*

Table 4.3 shows the mean loss ratios and corresponding coefficients of variation in the lognormal vulnerability function used in this test case. The vulnerability model is shown in Figure 4.1, where the dots represent the median loss ratios at a set of intensity levels.

Since there is no variability in the loss ratio, calculation of the loss ratios is straightforward in this case. Since the coefficients of variation in the vulnerability function are all zero, the lognormal distribution devolves into the degenerate distribution. The ground motion values at the location of the single asset are $[1.3, 0.044, 0.52, 1.0, 1.2]g$. Consider the first value of $PGA = 1.3g$. The vulnerability function for this case provides mean loss ratio values at intensity measure levels $1.2g$ and $1.4g$, but none at $1.3g$. The mean loss ratios at $1.2g$ and $1.4g$ are 0.67 and 0.80 respectively.

The mean loss ratio at $1.3g$ is obtained by interpolating between these two values. Linear interpolation gives a mean loss ratio of 0.735 for $PGA = 1.3g$.

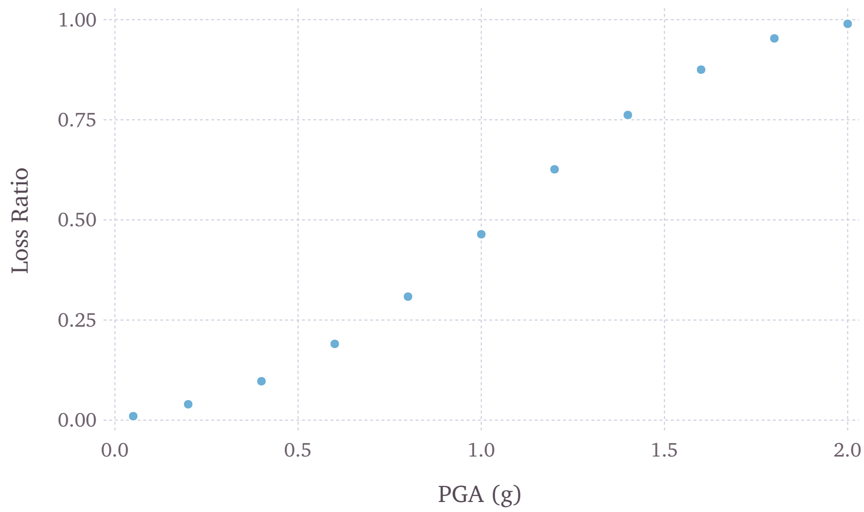

Figure 4.1 – *Vulnerability model with zero coefficients of variation*

Similar interpolation for the other ground motion values gives mean loss ratios of 0, 0.16, 0.5, and 0.67.

The mean loss ratio is simply obtained as the arithmetic mean of the five loss ratios as:

$$\frac{0.735+0.0+0.16+0.50+0.67}{5} = 0.413$$

The standard deviation of the loss ratio is computed as:

$$\sqrt{\frac{(0.735-0.413)^2+(0.0-0.413)^2+(0.16-0.413)^2+(0.50-0.413)^2+(0.67-0.413)^2}{5-1}}$$
$$= 0.320889$$

These numbers are multiplied by the asset value of $10,000$ to give the mean and standard deviation of loss for the scenario as $4,130$ and $3,208.89$ respectively. Table 4.4 shows the

Result	Expected	OpenQuake	Difference
Mean loss	4,130.00	4,130.00	0.00%
Std. loss	3,208.89	3,208.89	0.00%

Table 4.4 – *Results for scenario risk test case 1a*

comparison of the OpenQuake result with the expected result.

4.1.1.2 Case 1b

This test case is identical to Case 1a described above, except for the use of the Beta distribution for the vulnerability functions instead of the lognormal distribution. Since the coefficients of variation in the vulnerability function are all zero, once again the Beta distribution devolves into the degenerate distribution as in the previous case. The results for this test case should be exactly the same as in Case 1a. Table 4.5 shows the comparison of the OpenQuake result with the

Result	Expected	OpenQuake	Difference
Mean loss	4,130.00	4,130.00	0.00%
Std. loss	3,208.89	3,208.89	0.00%

Table 4.5 – *Results for scenario risk test case 1b*

expected result.

4.1.1.3 Case 1c

The purpose of this case is to test the correct sampling of the loss ratio from the prescribed distribution of the vulnerability function, given a specific intensity of ground motion. The 1,000 ground motion fields used in this test case are identical, i.e., variability in the ground motion is not considered in this case. However, in contrast to Case 1a, variability in the loss ratio *is* considered in the vulnerability function for this case. This permits us to compare the computed mean and standard deviation of the asset loss with the expected values, which are simply obtained through interpolation on the vulnerability function.

GMF #	Site	PGA (g)
1	1	0.5000
2	1	0.5000
3	1	0.5000
4	1	0.5000
⋮	⋮	⋮
1,000	1	0.5000

Table 4.6 – *1,000 identical ground motion fields at a single site*

Table 4.6 lists five of the one thousand identical ground motion values used in this test case.

PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
Mean LR	0.01	0.04	0.10	0.20	0.33	0.50	0.67	...	0.99
CoV LR	0.03	0.12	0.24	0.32	0.38	0.40	0.38	...	0.03

Table 4.7 – *Lognormal vulnerability function with nonzero coefficients of variation*

Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case. The vulnerability model is shown in Figure 4.2, where the dots represent the median loss ratios at a set of intensity levels, and the error bars depict the $16^{th} and 84^{th} percentiles of the conditional lognormal distribution at those intensity levels.$

The vulnerability function for this case provides mean loss ratio values and coefficients of variation at intensity measure levels $PGA = 0.4g$ and $0.6g$, but none at $0.5g$. Linear interpolation

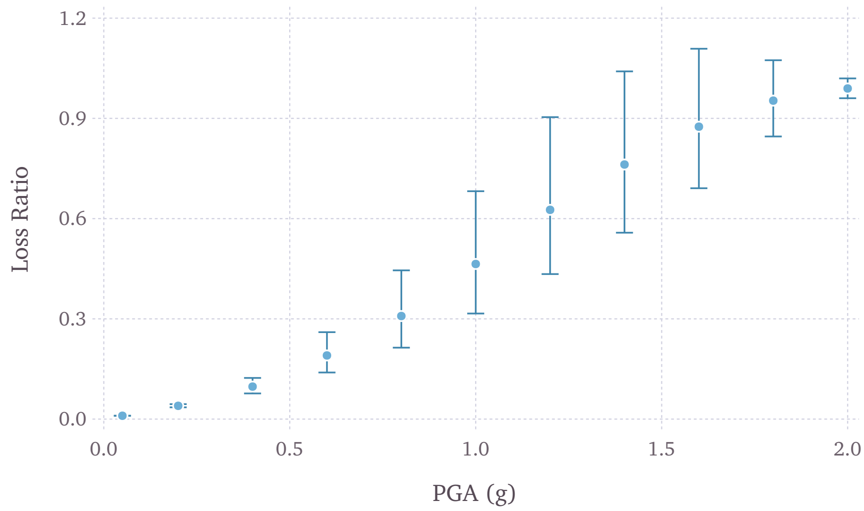

Figure 4.2 – *Vulnerability model with nonzero coefficients of variation*

gives a mean loss ratio of 0.15 for $PGA = 0.5g$. Similarly, the coefficients of variation of the loss ratio at $0.4g$ and $0.6g$ are 0.24 and 0.32 respectively. The coefficient of variation of the loss ratio for $PGA = 0.5g$ is obtained by linear interpolation as 0.28.

The loss ratio at $PGA = 0.5g$ follows a lognormal distribution with a mean of 0.15 and a standard deviation of $0.28 \times 0.15 = 0.042$.

Since there is no variability in the ground motion, the expected value of the mean loss ratio for the scenario is also 0.15, and the expected value of the standard deviation of the loss ratio is 0.042.

These numbers are multiplied by the asset value of $10,000$ to give the expected mean and standard deviation of loss for the scenario as $1,500$ and 420 respectively. Table 4.8 shows the

Result	Expected	OpenQuake	Difference
Mean loss	1,500.00	1,480.17	1.32%
Std. loss	420.00	410.18	2.34%

Table 4.8 – *Results for scenario risk test case 1c*

comparison of the OpenQuake result with the expected result.

4.1.1.4 Case 1d

This test case is identical to Case 1c described above, except for the use of the Beta CDF for the probabilistic fistribution of the vulnerability functions instead of the lognormal CDF.

Table 4.9 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case. The same one thousand identical ground motion values described earlier in Table 4.6 are used in this test case.

Since the mean loss ratios at the different intensity measure levels and the corresponding coefficients of variation in the vulnerability function are exactly the same as in Case 1c, the

PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
Mean LR	0.01	0.04	0.10	0.20	0.33	0.50	0.67	...	0.99
CoV LR	0.03	0.12	0.24	0.32	0.38	0.40	0.38	...	0.03

Table 4.9 – *Beta vulnerability function with nonzero coefficients of variation*

results for this test case should also be exactly the same as in Case 1c. Table 4.10 shows the

Result	Expected	OpenQuake	Difference
Mean loss	1,500.00	1,470.63	1.96%
Std. loss	420.00	1,345.84	-220.44%

Table 4.10 – *Results for scenario risk test case 1d*

comparison of the OpenQuake result with the expected result.

4.1.1.5 Case 1e

The purpose of this case is to test vulnerability functions that are specified as discrete probability mass functions rather than the parametric lognormal or beta distributions seen in the previous cases.

LR | PGA	0.05g	0.20g	0.40g	0.60g	1.00g	1.40g	1.60g	2.00g
0.000	0.995	0.950	0.490	0.300	0.140	0.030	0.010	0.004
0.005	0.004	0.030	0.380	0.400	0.300	0.100	0.030	0.006
0.050	0.001	0.015	0.080	0.160	0.240	0.300	0.100	0.010
0.200	0.000	0.004	0.020	0.080	0.160	0.260	0.300	0.030
0.450	0.000	0.001	0.015	0.030	0.100	0.180	0.300	0.180
0.800	0.000	0.000	0.010	0.020	0.040	0.100	0.180	0.390
1.000	0.000	0.000	0.005	0.010	0.020	0.030	0.080	0.380

Table 4.11 – *Vulnerability function specified using a discrete probability distribution. The values in each column specify the probability of occurrence of the corresponding loss ratio from the first column, for the ground motion intensity listed in the first row.*

The vulnerability function used in this test case is shown in Table 4.11. This vulnerability function specifies a set of loss ratios and the corresponding probabilities of occurrence for these loss ratios at different intensity measure levels.

The same identical ground motion values described earlier in Case 1c and shown in Table 4.6 are used in this test case. However, the total number of ground motion fields is increased to 10,000 for this case, since the spread in the loss ratio distribution is much larger for the vulnerability function used in this case.

The vulnerability function for this case provides probabilities of occurrence for a set of loss ratios 0.000, 0.005, 0.050, 0.200, 0.450, 0.800, 1.000 at intensity measure levels $PGA = 0.4g$ and $0.6g$, but not at $0.5g$. The specified set of probabilities for $PGA = 0.4g$ are 0.490, 0.380, 0.080, 0.020, 0.015, 0.010, 0.005, and those at $PGA = 0.6g$ are 0.300, 0.400, 0.160, 0.080, 0.030, 0.020, 0.010. Linear interpolation is used to obtain the probabilities of occurrence for the same set of loss ratios at $PGA = 0.5g$ as 0.395, 0.390, 0.120, 0.050, 0.0225, 0.015, 0.0075.

For the discrete random variable LR, which has the probability mass function (PMF): $lr_1 \mapsto p_1, \ldots, lr_n \mapsto p_n$, the mean and standard deviation are calculated as:

$$\mu_{LR} = \sum_{i=1}^{n} p_i \cdot lr_i \tag{4.1}$$

$$\sigma_{LR} = \sqrt{\sum_{i=1}^{n} p_i \cdot lr_i^2 - \mu_{LR}^2} \tag{4.2}$$

Based on the above equations, the expected values of the mean and standard deviation of the loss ratio for our case are calculated as 0.047575 and 0.147318 respectively. These numbers are multiplied by the asset value of $10,000$ to give the expected mean and standard deviation of loss for the scenario as 475.75 and $1,473.18$ respectively. Table 4.12 shows the comparison of the OpenQuake result with the expected result.

Result	Expected	OpenQuake	Difference
Mean loss	475.75	494.98	-4.04%
Std. loss	1,473.18	1,510.47	-2.53%

Table 4.12 – *Results for scenario risk test case 1e*

4.1.1.6 Case 1f

Variability in the ground motion is considered in all cases starting from Case 1f. Ten thousand ground motion fields are generated for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008).

The purpose of this case is to test the computation of the mean and standard deviation of the loss, given variability in both the ground motion values and in the lognormal vulnerability functions.

GMF #	Site	PGA (g)
1	1	1.3495
2	1	0.5393
3	1	0.5240
4	1	1.0385
⋮	⋮	⋮
10,000	1	0.1327

Table 4.13 – *10,000 simulated ground motion fields*

Table 4.13 lists five of the ten thousand ground motion values generated by OpenQuake. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Since the mean loss ratios in the vulnerability function are not a linear function of the intensity measure levels, an analytical solution for the mean and standard deviation of loss for the scenario cannot be found as in the previous cases. Thus, in order to check the OpenQuake results, an alternate implementation of the calculator algorithm in the programming language Julia is used for comparison. In order to provide a representative baseline for the comparison, one million ground motion fields are used in the Julia calculation.

The mean and standard deviation of the logarithm of the ground motion calculated at the location of the asset as obtained by using the Boore and Atkinson (2008) equation are -0.648 and 0.564 respectively. Assuming a lognormal distribution for the variability in the ground motion, one million motion values are generated using Julia with these logarithmic mean and standard deviation values.

The mean loss ratio and standard deviation of loss ratio for each simulated ground motion value are obtained through interpolation on the mean loss ratios and corresponding coefficients of variation provided by the vulnerability function. Using the interpolated mean and standard deviation of loss ratios, one loss ratio is sampled for each ground motion value, assuming a lognormal distibution.

The mean and standard deviation of loss ratio for the scenario are estimated simply as the mean and standard deviation of the ten thousand simulated loss ratios. These numbers are then multiplied by the asset value of $10,000$ to give the expected mean and standard deviation of loss for the scenario. Table 4.14 shows the comparison of the OpenQuake result with the expected

Result	Julia	OpenQuake	Difference
Mean loss	2,404.92	2,370.74	1.42%
Std. loss	2,419.63	2,401.76	0.74%

Table 4.14 – *Results for scenario risk test case 1f*

result.

4.1.1.7 Case 1g

This test case is identical to Case 1f described above, except for the use of the Beta distribution for the vulnerability functions instead of the lognormal distribution used in the previous case. Table 4.15 shows the comparison of the OpenQuake result with the expected result.

Result	Julia	OpenQuake	Difference
Mean loss	2,400.23		%
Std. loss	2,417.74		%

Table 4.15 – *Results for scenario risk test case 1g*

4.1.1.8 Case 1h

This test case repeats the exercise from Case 1f and Case 1g, using the discrete probability vulnerability functions instead of the parametric lognormal or Beta distribution based functions used in the previous two cases.

In this case, for each simulated ground motion value, the probabilities of occurrence of the set of loss ratios used by the vulnerability function are obtained through interpolation as described earlier in Case 1c. Using the set of loss ratios and the corresponding interpolated probabilities, one loss ratio is sampled for each ground motion value.

The mean and standard deviation of loss ratio for the scenario are estimated simply as the mean and standard deviation of the ten thousand simulated loss ratios. The OpenQuake values are compared with the alternate implementation of the algorithm in Julia. Table 4.16 shows the comparison of the OpenQuake result with the expected result.

Result	Julia	OpenQuake	Difference
Mean loss	819.96	823.21	-0.40%
Std. loss	2,006.80	2,004.00	-0.14%

Table 4.16 – *Results for scenario risk test case 1h*

4.1.1.9 Case 2a

In addition to computing direct structural losses, OpenQuake also provides support for computing losses incurred for the following other loss types:

- Non-structural losses
- Contents losses
- Downtime, or business interruption losses
- Occupant fatalities

PGA	0.005g	0.15g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
Mean LR	0.01	0.05	0.12	0.24	0.40	0.60	0.80	...	1.00
CoV LR	0.03	0.12	0.24	0.32	0.32	0.24	0.12	...	0.00

Table 4.17 – *Lognormal vulnerability function for non-structural components*

The purpose of this case is to test the calculation of mean and standard deviation of non-structural losses for an asset. The replacement value of the non-structural components for the asset used in this case is $15,000$. Table 4.17 shows the mean loss ratios and corresponding coefficients of variation in the non-structural components vulnerability function used in this test case. Table 4.18 shows the comparison of the OpenQuake result with the expected result.

4.1.1.10 Case 2b

The purpose of this case is to test the calculation of mean and standard deviation of the contents losses for an asset. The replacement value of the contents for the asset used in this case is $5,000$. Table 4.19 shows the mean loss ratios and corresponding coefficients of variation in the contents vulnerability function used in this test case.

Result	Julia	OpenQuake	Difference
Mean loss	4,304.94	4,246.21	1.36%
Std. loss	3,868.27	3,832.43	0.93%

Table 4.18 – *Results for scenario risk test case 2a*

PGA	**0.005g**	**0.15g**	**0.40g**	**0.60g**	**0.80g**	**1.00g**	**1.20g**	**...**	**2.00g**
Mean LR	0.02	0.10	0.33	0.66	0.90	0.98	1.00	...	1.00
CoV LR	0.03	0.12	0.24	0.24	0.12	0.03	0.00	...	0.00

Table 4.19 – *Lognormal vulnerability function for contents*

Table 4.20 shows the comparison of the OpenQuake result with the expected result.

4.1.1.11 Case 2c

The purpose of this case is to test the calculation of mean and standard deviation of downtime, or business-interruption losses for an asset. The loss due to downtime, or business-interruption for the asset used in this case is $2,000/month$. Downtime losses are usually specified per unit time the asset will be unavailable for occupancy or use. Table 4.21 shows the mean loss ratios and corresponding coefficients of variation for the downtime vulnerability function used in this test case.

Table 4.22 shows the comparison of the OpenQuake result with the expected result.

4.1.1.12 Case 2d

The purpose of this case is to test the calculation of mean and standard deviation of occupant fatalities for an asset. The number of occupants for the asset used in this case are 2 (day), 4 (transit), and 6 (night). Table 4.21 shows the mean loss ratios and corresponding coefficients of variation for the occupants fatality vulnerability function used in this test case.

Table 4.24 shows the comparison of the OpenQuake result with the expected result.

4.1.1.13 Case 3a

There are several ways by which the replacement value of an asset can be specified in the exposure model. The different options are listed below:

- Specify the aggregate value of each asset
- Specify the value per unit, and provide the number of units in each asset
- Specify the value per unit area, and provide the aggregate area of each asset
- Specify the value per unit area, specify the area per unit, and provide the number of units in each asset

Result	Julia	OpenQuake	Difference
Mean loss	2,819.17	2,799.75	0.69%
Std. loss	1,548.18	1,537.22	0.71%

Table 4.20 – *Results for scenario risk test case 2b*

PGA	0.005g	0.15g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
Mean LR	0.01	0.04	0.10	0.20	0.33	0.50	0.67	...	0.99
CoV LR	0.03	0.12	0.24	0.32	0.38	0.40	0.38	...	0.03

Table 4.21 – *Lognormal vulnerability function for downtime*

Result	Julia	OpenQuake	Difference
Mean loss	483.65	478.04	1.16%
Std. loss	480.02	477.50	0.52%

Table 4.22 – *Results for scenario risk test case 2c*

This case tests the computation of the mean and standard deviation of the loss when the aggregate asset value is provided in the exposure model. The vulnerability function used is the same as in Case 1f and shown in Table 4.7. The aggregate asset value in this case is 20,000. Table 4.25 shows the comparison of the OpenQuake result with the expected result.

4.1.1.14 Case 3b

This case tests the computation of the mean and standard deviation of the loss when the value of the assets is specified per unit, and the number of units in each asset are provided in the exposure model. The vulnerability function used is the same as in Case 1f and shown in Table 4.7. The asset has two units, and the value per unit is 7,500. The aggregate asset value in this case is 15,000. Table 4.26 shows the comparison of the OpenQuake result with the expected result.

4.1.1.15 Case 3c

This case tests the computation of the mean and standard deviation of the loss when the value of the assets is specified per unit area, and the aggregate area of each asset is provided in the exposure model. The vulnerability function used is the same as in Case 1f and shown in Table 4.7. The asset has an aggregate area of 1,000 sq. units, and the value per unit area is 5. The aggregate asset value in this case is 5,000. Table 4.27 shows the comparison of the OpenQuake result with the expected result.

4.1.1.16 Case 3d

This case tests the computation of the mean and standard deviation of the loss when the value of the assets is specified per unit area, the area is specified per unit, and the number of units in each asset are provided in the exposure model. The vulnerability function used is the same as in Case 1f and shown in Table 4.7. The asset has three units, the area per unit is 400 sq. units, and the value per unit area is 10. The aggregate asset value in this case is 12,000. Table 4.28 shows

PGA	0.005g	0.15g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
Mean LR	0.0001	0.0004	0.0010	0.0020	0.0033	0.0050	0.0067	...	0.0099
CoV LR	0.03	0.12	0.24	0.32	0.38	0.40	0.38	...	0.03

Table 4.23 – *Lognormal vulnerability function for occupants fatality*

Result	Julia	OpenQuake	Difference
Mean loss	1.45×10^{-2}	9.56×10^{-3}	34.12%
Std. loss	1.44×10^{-2}	9.55×10^{-3}	33.80%

Table 4.24 – *Results for scenario risk test case 2d*

Result	Julia	OpenQuake	Difference
Mean loss	4,809.84	4,741.48	1.42%
Std. loss	4,839.26	4,803.54	0.74%

Table 4.25 – *Results for scenario risk test case 3a*

Result	Julia	OpenQuake	Difference
Mean loss	3,607.38	3,556.11	1.42%
Std. loss	3,629.45	3,602.65	0.74%

Table 4.26 – *Results for scenario risk test case 3b*

Result	Julia	OpenQuake	Difference
Mean loss	1,202.46	1,185.37	1.42%
Std. loss	1,209.82	1,200.88	0.74%

Table 4.27 – *Results for scenario risk test case 3c*

Result	Julia	OpenQuake	Difference
Mean loss	2,885.90	2,844.89	1.42%
Std. loss	2,903.56	2,882.12	0.74%

Table 4.28 – *Results for scenario risk test case 3d*

Site	Taxonomy	Latitude	Longitude	Comment
1	tax1	38.113	-122.000	On fault midpoint, along strike
2	tax2	38.113	-122.114	10 km west of fault, at midpoint
3	tax1	38.113	-122.570	50 km west of fault, at midpoint
4	tax3	38.000	-122.000	South end of fault
5	tax1	37.910	-122.000	10 km south of fault, along strike
6	tax2	38.225	-122.000	North end of fault
7	tax1	38.113	-121.886	10 km east of fault, at midpoint

Table 4.29 – *Asset sites and taxonomies for the multiple-asset, multiple-taxonomy test cases*

the comparison of the OpenQuake result with the expected result.

4.1.2 Multiple asset tests

The multiple asset test cases are designed to test the loss aggregation functions of the scenario risk calculator, such as:

- portfolio loss computation for a given ground motion field
- calculation of mean and standard deviation of portfolio scenario loss
- loss correlation between assets of the same taxonomy

4.1.2.1 Case 4a

The purpose of this case is to test the basic elements of a scenario risk calculation involving multiple assets, such as the computation of the mean and standard deviation of the total loss for a portfolio of assets.

The list of assets and their taxonomies are shown in Table 4.29.

Five precomputed ground motion fields are used as the starting point for this case. These ground motion fields take into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied.

GMF #	Site 3	Site 2	Site 5	Site 4	Site 1	Site 6	Site 7
1	0.15g	0.17g	0.21g	0.56g	0.25g	0.38g	0.14g
2	0.05g	0.21g	0.18g	0.69g	0.94g	0.72g	0.43g
3	0.05g	0.18g	0.06g	0.58g	0.46g	0.24g	0.22g
4	0.15g	0.46g	0.72g	0.79g	0.81g	0.29g	0.51g
5	0.15g	0.48g	0.95g	1.70g	1.70g	0.63g	0.25g

Table 4.30 – *Five precomputed spatially correlated ground motion fields (PGA). The sites are sorted first by longitude, then by latitude.*

Table 4.30 lists the ground motion fields used in this test case.

Table 4.31 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Taxonomy	PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	...	2.00g
tax1	**Mean LR**	0.01	0.04	0.10	0.20	0.33	0.50	...	0.99
	CoV LR	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
tax2	**Mean LR**	0.01	0.02	0.05	0.11	0.18	0.26	...	0.51
	CoV LR	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
tax3	**Mean LR**	0.01	0.04	0.09	0.18	0.28	0.47	...	0.91
	CoV LR	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0

Table 4.31 – *Lognormal vulnerability functions for three building typologies*

Since there is no variability in the loss ratio, calculation of the loss ratios is straightforward in this case. Consider asset $a3$, which has the taxonomy $tax1$. The ground motion values at the location of the single asset are $[0.15, 0.05, 0.05, 0.15, 0.15]g$. Consider the first value of $PGA = 0.15g$. The vulnerability function for this taxonomy provides mean loss ratio values at intensity measure levels $0.05g$ and $0.20g$, but none at $0.15g$. The mean loss ratios at $0.05g$ and $0.20g$ are 0.01 and 0.04 respectively.

The mean loss ratio at $0.15g$ is obtained by interpolating between these two values. Linear interpolation gives a mean loss ratio of 0.03 for $PGA = 0.15g$.

Similar interpolation for the other ground motion values gives mean loss ratios of 0.01, 0.01, 0.03, and 0.03 respectively. These numbers are multiplied by the asset value for $a1$ of $10,000$ to give loss values for asset $a1$ of $[300, 100, 100, 300, 300]$.

Repeating this exercise for the six other assets, using the appropriate vulnerability function for each taxonomy, we get the following loss values for the five ground motion fields:

- $a1$: $[550, 4490, 1300, 3385, 9300]$
- $a2$: $[180, 215, 186.67, 680, 740]$
- $a3$: $[1800, 2585, 1900, 3235, 9300]$
- $a4$: $[1620, 2250, 1710, 2750, 8200]$
- $a5$: $[430, 360, 120, 2780, 4575]$
- $a6$: $[470, 1520, 260, 335, 1205]$
- $a7$: $[280, 1150, 460, 1550, 550]$

The portfolio losses are obtained simply as the sum of all the individual asset losses for each ground motion field. The portfolio losses are: $[5330, 12570, 5936.67, 14715, 33870]$.

Now, the mean and standard deviation of the scenario loss can be calculated for each of the individual assets, as well as for the portfolio. The expected values of these statistics are provided in Table 4.32, and the OpenQuake results for the same are also provided in the same table for comparison.

4.1.2.2 Case 4b

This case is designed to test the computation of the mean and standard deviation of individual asset losses and also for the portfolio, when the vulnerability models of different assets of the same taxonomy are treated as uncorrelated. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter ‘asset_correlation’ to zero.

Asset	Result	Expected	OpenQuake	Difference
a1	Mean loss	3,805.00	3,805.00	0.00%
	Std. loss	3,453.65	3,453.65	0.00%
a2	Mean loss	400.33	400.33	0.00%
	Std. loss	283.78	283.78	0.00%
a3	Mean loss	3,764.00	3,764.00	0.00%
	Std. loss	3,148.37	3,148.37	0.00%
a4	Mean loss	3,306.00	3,306.00	0.00%
	Std. loss	2,773.32	2,773.32	0.00%
a5	Mean loss	1,653.00	1,653.00	0.00%
	Std. loss	1,957.41	1,957.41	0.00%
a6	Mean loss	758.00	758.00	0.00%
	Std. loss	567.96	567.96	0.00%
a7	Mean loss	798.00	798.00	0.00%
	Std. loss	532.33	532.33	0.00%
Total	Mean loss	14,484.33	14,484.33	0.00%
	Std. loss	11,580.01	11,580.01	0.00%

Table 4.32 – *Results for scenario risk test case 4a*

Ten thousand ground motion fields are generated for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied. These ground motion fields are also used for the corresponding calculation in Julia.

GMF #	Site 3	Site 2	Site 5	Site 4	Site 1	Site 6	Site 7
1	0.15g	0.17g	0.21g	0.56g	0.25g	0.38g	0.14g
2	0.05g	0.21g	0.18g	0.69g	0.94g	0.72g	0.43g
3	0.05g	0.18g	0.06g	0.58g	0.46g	0.24g	0.22g
4	0.15g	0.46g	0.72g	0.79g	0.81g	0.29g	0.51g
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
10,000	0.04g	0.15g	0.32g	0.32g	0.77g	0.95g	0.16g

Table 4.33 – *10,000 simulated spatially correlated ground motion fields (PGA). The sites are sorted first by longitude, then by latitude.*

Table 4.33 lists five of the ten thousand ground motion fields generated by the OpenQuake scenario hazard calculator. The list of assets in the exposure model used for the test cases involving asset correlation (Cases 4b, 4c, and 4d) is given in Table 4.34. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Site	Taxonomy	Latitude	Longitude	Comment
1	tax1	38.113	-122.000	On fault midpoint, along strike
2	tax1	38.113	-122.114	10 km west of fault, at midpoint
3	tax1	38.113	-122.570	50 km west of fault, at midpoint
4	tax1	38.000	-122.000	South end of fault
5	tax1	37.910	-122.000	10 km south of fault, along strike
6	tax1	38.225	-122.000	North end of fault
7	tax1	38.113	-121.886	10 km east of fault, at midpoint

Table 4.34 – *Asset sites and taxonomies for the multiple-asset, single-taxonomy test cases*

Since the sampled loss ratios conditional on a given ground motion field for different assets of the same taxonomy are assumed to be uncorrelated in this case, a loss ratio is sampled independently for each asset from the univariate lognormal distribution for that asset for each ground motion field. Table 4.35 shows the comparison of the OpenQuake results with the results

Asset	Result	Julia	OpenQuake	Difference
a1	Mean loss	2,381.45	2,390.58	0.38%
	Std. loss	2,372.78	2,404.29	1.32%
a2	Mean loss	617.07	617.52	0.07%
	Std. loss	637.74	648.60	1.69%
a3	Mean loss	147.53	147.63	0.07%
	Std. loss	125.24	124.91	0.27%
a4	Mean loss	2,398.26	2,399.53	0.05%
	Std. loss	2,392.45	2,409.41	0.71%
a5	Mean loss	622.40	621.42	0.16%
	Std. loss	635.35	635.06	0.05%
a6	Mean loss	2,086.46	2,089.41	0.14%
	Std. loss	2,126.51	2,180.91	2.53%
a7	Mean loss	621.92	621.47	0.07%
	Std. loss	635.27	650.05	2.30%
Total	Mean loss	8,875.09	8,887.56	0.14%
	Std. loss	6,060.60	6,121.63	1.00%

Table 4.35 – *Results for scenario risk test case 4b*

from Julia.

4.1.2.3 Case 4c

This case is designed to test the computation of the mean and standard deviation of individual asset losses and also for the portfolio, when the vulnerability models of different assets of the same taxonomy are treated as fully correlated. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter 'asset_correlation' to one.

Ten thousand ground motion fields are generated for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied. These ground motion fields are also used for the corresponding calculation in Julia.

Table 4.33 lists five of the ten thousand ground motion fields generated by the OpenQuake scenario hazard calculator. The list of assets in the exposure model used for this case is given in Table 4.34. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Since the sampled loss ratios conditional on a given ground motion field for different assets of the same taxonomy are assumed to be fully correlated in this case, a single *epsilon*, ε, is sampled from the standard normal distribution for each taxonomy. The parameters m and s from the vulnerability model are converted to the parameters μ and σ of the corresponding normal distribution, and the sampled loss ratio is obtained simply as $\exp(\mu + \varepsilon * \sigma)$. Table 4.36 shows

Asset	Result	Julia	OpenQuake	Difference
a1	Mean loss	2,401.43	2,394.67	0.28%
	Std. loss	2,427.09	2,430.56	0.14%
a2	Mean loss	617.38	619.11	0.28%
	Std. loss	635.56	660.50	3.85%
a3	Mean loss	147.71	147.78	0.05%
	Std. loss	125.91	126.79	0.70%
a4	Mean loss	2,409.00	2,389.95	0.79%
	Std. loss	2,409.27	2,400.60	0.36%
a5	Mean loss	620.52	621.46	0.15%
	Std. loss	631.77	629.01	0.44%
a6	Mean loss	2,089.45	2,088.19	0.06%
	Std. loss	2,165.38	2,162.85	0.12%
a7	Mean loss	622.99	620.99	0.32%
	Std. loss	644.46	646.60	0.33%
Total	Mean loss	8,908.47	8,882.14	0.30%
	Std. loss	6,676.47	6,477.06	3.03%

Table 4.36 – *Results for scenario risk test case 4c*

the comparison of the OpenQuake results with the results from Julia.

4.1.2.4 Case 4d

This case is designed to test the computation of the mean and standard deviation of individual asset losses and also for the portfolio, when the vulnerability models of different assets of the same taxonomy are treated as partially correlated, with a correlation coefficient of 0.5. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter ‘asset_correlation’ to 0.5.

As in the previous cases for zero and full correlation, ten thousand ground motion fields are generated for the given rupture, and these ground motion fields are also used for the corresponding calculation in Julia. The list of assets in the exposure model used for this case is given in Table 4.34. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Since the sampled loss ratios conditional on a given ground motion field for different assets of the same taxonomy are assumed to be correlated in this case, we proceed by first generating a vector of *epsilons* for each taxonomy from the multivariate standard normal distribution which has the symmetric covariance matrix with 1.0 as the diagonal elements and $\rho = 0.5$ as the off-diagonal elements.

Now, for each asset of that taxonomy, the parameters m and s are obtained for the ground motion value at the location of the asset through interpolation on the specified vulnerability model. Each asset of a particular taxonomy is also assigned a value of ε from the vector of *epsilons* for that taxonomy sampled as described above. The parameters m and s are then converted to the parameters μ and σ of the corresponding normal distribution, and the sampled loss ratio is obtained simply as $\exp(\mu + \varepsilon * \sigma)$.

After the asset event loss tables are compiled by sampling correlated loss values as described above, the rest of the calculation concerning the derivation of loss curves and average loss proceeds as in previous cases. Table 4.37 shows the comparison of the OpenQuake result with

Asset	Result	Julia	OpenQuake	Difference
a1	Mean loss	2,392.95	2,395.46	0.10%
	Std. loss	2,389.29	2,419.20	1.24%
a2	Mean loss	620.15	619.49	0.11%
	Std. loss	672.44	656.71	2.37%
a3	Mean loss	147.91	147.72	0.13%
	Std. loss	126.79	126.65	0.11%
a4	Mean loss	2,397.75	2,393.09	0.19%
	Std. loss	2,374.34	2,398.43	1.01%
a5	Mean loss	622.79	624.65	0.30%
	Std. loss	650.77	650.28	0.08%
a6	Mean loss	2,082.74	2,087.26	0.22%
	Std. loss	2,171.86	2,178.06	0.28%
a7	Mean loss	622.92	623.61	0.11%
	Std. loss	630.64	649.76	2.99%
Total	Mean loss	8,887.21	8,891.27	0.05%
	Std. loss	6,331.38	6,308.61	0.36%

Table 4.37 – *Results for scenario risk test case 4d*

the results from Julia.

4.1.3 Insurance tests

4.1.3.1 Case 5a

In addition to calculating individual asset losses and portfolio losses for a scenario event, OpenQuake also calculates the insured losses for individual assets and for the portfolio, if requested. Each loss type can be assigned a deductible and an insurance limit, specified in either relative or absolute terms with respect to the replacement value of the asset.

Given an asset loss *loss*, and a deductible component of insurance *deductible*, and an insurance limit *limit*, the insured loss is zero if the asset loss is below the deductible. For losses above the deductible amount, insurance pays the difference up to the limit. The insured loss is thus the smaller of the difference and the insurance limit. The equation used for computing the insured loss is presented below:

$$insured\_loss = min(max(loss - deductible, 0.0), limit - deductible) \tag{4.3}$$

The insured asset losses are collected for each of the ground motion fields, and finally the mean and standard deviation of the asset insured losses are calculated. The portfolio insured loss mean and standard deviation are also computed.

The input models for this test case are based on those used earlier in Case 1f. The deductible component of insurance is $0.1\times$ the cost of replacement of the asset. The insurance limit is capped at $0.8\times$ the cost of replacement of the asset. Table 4.38 shows the comparison of the

Result	Julia	OpenQuake	Difference
Mean insured loss	1,411.89	1,386.42	1.82%
Std. insured loss	1,997.08	1,981.63	0.78%

Table 4.38 – *Results for scenario risk test case 5a*

OpenQuake result with the expected result.

4.1.4 Calculation with logic-trees

4.1.4.1 Case 6a

The OpenQuake scenario risk calculator allows the user to employ more than one ground motion prediction equation (GMPE) for computing the ground motion fields used for the loss calculation. The mean and standard deviation of the individual asset losses, portfolio losses, and insured losses (if any), are calculated and output for each GMPE branch independently. No sampling is involved, and any branch weights assigned to the different GMPE branches are ignored.

A single asset is used in this test case. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

The two ground motion prediction equations used are Boore and Atkinson (2008), and Chiou and Youngs (2008). Ten thousand ground motion fields are generated using OpenQuake for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion, for both GMPE branches. The rest of the loss calculation procedure follows the

Branch	Result	Julia	OpenQuake	Difference
BA2008	Mean loss	2,397.78	2,370.74	1.13%
	Std. loss	2,413.91	2,401.77	0.50%
CY2008	Mean loss	2,922.36	2,878.58	1.50%
	Std. loss	2,691.42	2,652.25	1.46%

Table 4.39 – *Results for scenario risk test case 6a*

same steps as described earlier in Case 1f. Table 4.39 shows the comparison of the OpenQuake result with the expected result.

4.2 Scenario Damage Calculator

The tests for the scenario damage calculator assume the correct computation of the ground motion fields at the locations of the assets in the exposure model. Thus, the risk quality assurance tests implicitly rely on the acceptance tests for the scenario hazard calculator.

The rupture model used for the tests comprises a magnitude $M6.7$ rupture on a vertical strike-slip fault, the same as is used in the tests for the scenario risk calculator.

Details of the rupture are repeated below for convenience:

Fault type: Strike slip
Fault dip: 90°
Fault plane depths: 0–20 km
Fault coordinates:
South end: $38.0000°N$, $122.0000°W$
North end: $38.2248°N$, $122.0000°W$
Rupture magnitude: 6.7
Rupture hypocenter: $38.1124°N$, $122.0000°W$
Hypocenter depth: 10 km

The complete collection of input models and job configuration files used in these test cases can be accessed here: https://github.com/gem/oq-risklib/tree/master/openquake/qa_tests_data/scenario_damage

4.2.1 Single asset tests

The single asset test cases are designed to test the basic elements of the scenario damage calculator, such as:

- interpolation of the discrete fragility functions
- damage distribution computation for a given set of ground motion fields
- extraction of the probability of collapse

The location and taxonomy of the single asset in the exposure model used for the single-asset test cases for the scenario risk calculator are given in Table 4.1.

4.2.1.1 Case 1a

Test Case 1a uses a set of five precomputed ground motion values to test the correct interpolation of the damage state exceedance probabilities of the discrete fragility function at intermediate intensity measure levels.

Table 4.2 lists the five ground motion values used in this test case.

LS | PGA	0.2g	0.4g	0.6g	0.8g	1.0g	1.2g	1.4g	...	5.0g
ds1	0.000	0.152	0.846	0.993	1.000	1.000	1.000	...	1.000
ds2	0.000	0.014	0.129	0.350	0.576	0.747	0.857	...	1.000
ds3	0.000	0.008	0.085	0.196	0.325	0.450	0.561	...	0.993
ds4	0.000	0.006	0.067	0.171	0.263	0.354	0.438	...	0.951

Table 4.40 – *Discrete fragility function with zero no damage limit*

Table 4.40 shows the set of ground motion intensity levels and corresponding probabilities of exceedance for the four damage states for the discrete fragility function used in this test case. The fragility model is shown in Figure 4.3.

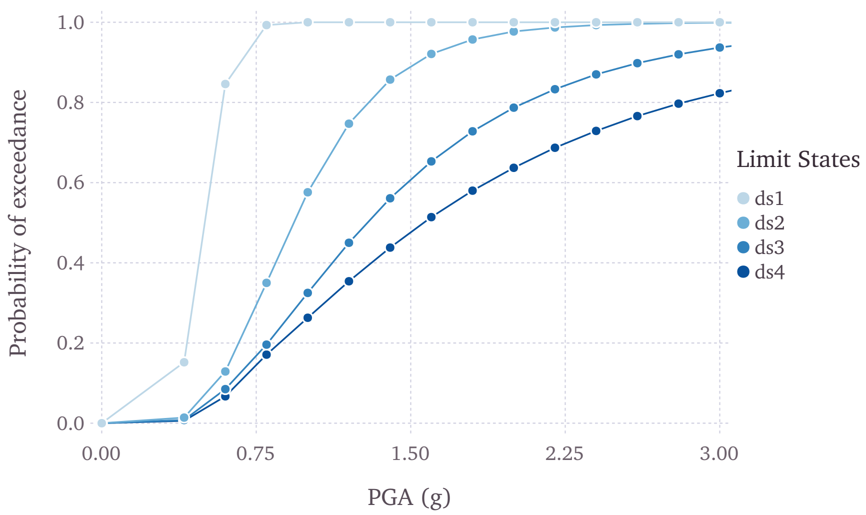

Figure 4.3 – *Discrete fragility model with four damage states*

The ground motion values at the location of the single asset are $[1.3, 0.044, 0.52, 1.0, 1.2]g$. Consider the first value of $PGA = 1.3g$. The discrete fragility function for this case provides damage state probabilities of exceedance at intensity measure levels $1.2g$ and $1.4g$, but none at $1.3g$. The exceedance probabilities at $1.2g$ and $1.4g$ corresponding to the discrete damage states $[ds_1, ds_2, ds_3, ds_4]$ are $[1.000, 0.747, 0.450, 0.354]$ and $[1.000, 0.857, 0.561, 0.438]$ respectively.

The exceedance probabilities at $1.3g$ are obtained by interpolating between these two sets of values. Linear interpolation gives exceedance probabilities of $[1.000, 0.802, 0.5055, 0.396]$ for $PGA = 1.3g$. The probabilities of damage state occurrence are given by the pairwise differences of the exceedance probabilities as $[1.000 - 0.802, 0.802 - 0.5055, 0.5055 - 0.396, 0.396] =$

$[0.198, 0.2965, 0.1095, 0.396]$. These four damage state probabilities sum up to one, indicating that the probability of observing no damage is zero.

Similar interpolation at the other four ground motion intensity levels gives the following sets of damage state probabilities:

- GMF1: $[0.000, 0.198, 0.2965, 0.1095, 0.396]$
- GMF2: $[1.000, 0.000, 0.000, 0.000, 0.000]$
- GMF3: $[0.4316, 0.4854, 0.0288, 0.0116, 0.0426]$
- GMF4: $[0.000, 0.424, 0.251, 0.062, 0.263]$
- GMF5: $[0.000, 0.253, 0.297, 0.096, 0.354]$

The mean and standard deviation of the four damage state probabilities and also the probability of observing no damage is now calculated using the above set of probabilities collected from each of the five ground motion simulations. Table 4.41 shows the comparison of the OpenQuake

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	0.2863	0.2863	0.00%
		Std.	0.4406	0.4406	0.00%
	ds1	Mean	0.2721	0.2721	0.00%
		Std.	0.1927	0.1927	0.00%
	ds2	Mean	0.1747	0.1747	0.00%
		Std.	0.1478	0.1478	0.00%
	ds3	Mean	0.0558	0.0558	0.00%
		Std.	0.0490	0.0490	0.00%
	ds4	Mean	0.2111	0.2111	0.00%
		Std.	0.1805	0.1805	0.00%

Table 4.41 – *Results for scenario damage test case 1a*

result with the expected result.

4.2.1.2 Case 1b

Whereas the previous case was concerned with checking the correct implementation and usage of *discrete* fragility functions, the purpose of this case is to verify the correct calculation of damage distribution statistics for a scenario using *continuous* (lognormal CDF) fragility functions.

LS	Mean IML	Std. IML
ds1	0.50	0.40
ds2	1.00	0.80
ds3	1.50	1.20
ds4	2.00	1.60

Table 4.42 – *Fragility function with zero no damage limit*

Table 4.42 shows the mean and standard deviation of the ground motion intensity level for the four damage states, which are the parameters for the lognormal fragility function used in this test case. The fragility model is shown in Figure 4.4.

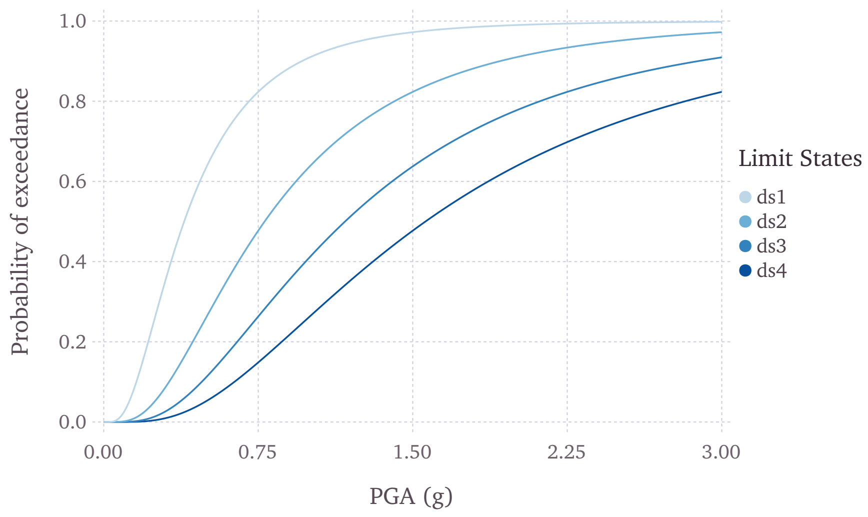

Figure 4.4 – *Continuous (lognormal) fragility model with four damage states*

The set of five precomputed ground motion values described in Table 4.2 are used in this case. The ground motion values at the location of the single asset are $[1.3, 0.044, 0.52, 1.0, 1.2]g$. Consider the first value of $PGA = 1.3g$. The exceedance probability for damage state ds_1 is obtained by employing the equation for the cumulative distribution function for the lognormal distribution, using the mean, $m_1 = 0.5g$, and standard deviation, $s_1 = 0.4g$, specified by the fragility function for that damage state. The equations are given below:

$$\mu_1 = \ln\left(\frac{m_1}{\sqrt{1+\frac{s_1^2}{m_1^2}}}\right) = -0.940 \tag{4.4}$$

$$\sigma_1 = \sqrt{\ln\left(1+\frac{s_1^2}{m_1^2}\right)} = 0.703 \tag{4.5}$$

$$p.o.e.(ds_1) = \frac{1}{2} + \frac{1}{2} erf\left[\frac{\ln 1.3 - \mu_1}{\sqrt{2}\sigma_1}\right] = 0.956 \tag{4.6}$$

Next, the exceedance probabilities for the other damage states for the first ground motion are obtained in a similar manner. We have, for the first ground motion:

- $p.o.e.(ds_1) = 0.956$
- $p.o.e.(ds_2) = 0.766$
- $p.o.e.(ds_3) = 0.559$
- $p.o.e.(ds_4) = 0.397$

Thus we have exceedance probabilities of $[0.956, 0.766, 0.559, 0.397]$ for $PGA = 1.3g$. The probabilities of damage state occurrence are given by the pairwise differences of the exceedance probabilities as $[0.956-0.766, 0.766-0.559, 0.559-0.397, 0.397] = [0.190, 0.207, 0.162, 0.397]$. The probability of observing no damage is the remainder of the probability after summing up the probabilities for the four damage states, i.e., $1.0-(0.190+0.207+0.162+0.397) = 0.044$.

This procedure is repeated for the other four ground motion fields to give the following sets of damage state probabilities:

- GMF1: $[0.0436, 0.191, 0.207, 0.162, 0.397]$
- GMF2: $[0.9991, 0.0009, 0.000, 0.000, 0.000]$
- GMF3: $[0.342, 0.376, 0.157, 0.0652, 0.059]$
- GMF4: $[0.091, 0.272, 0.226, 0.148, 0.263]$
- GMF5: $[0.055, 0.215, 0.215, 0.160, 0.354]$

The mean and standard deviation of the four damage state probabilities and also the probability of observing no damage is now calculated using the above set of probabilities collected from each of the five ground motion simulations. Table 4.43 shows the comparison of the OpenQuake result with the expected result.

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	0.3061	0.3061	0.00%
		Std.	0.4061	0.4061	0.00%
	ds1	Mean	0.2111	0.2111	0.00%
		Std.	0.1376	0.1376	0.00%
	ds2	Mean	0.1613	0.1613	0.00%
		Std.	0.0939	0.0939	0.00%
	ds3	Mean	0.1069	0.1069	0.00%
		Std.	0.0719	0.0719	0.00%
	ds4	Mean	0.2146	0.2146	0.00%
		Std.	0.1770	0.1770	0.00%

Table 4.43 – *Results for scenario damage test case 1b*

4.2.1.3 Case 1c

Test Case 1c repeats the exercise from Case 1a, with the difference that the discrete fragility function specifies a minimum ground motion intensity, below which the probability of exceedance for all damage states is assumed to be zero.

Table 4.44 lists the five ground motion values used in this test case, and Table 4.40 the discrete fragility function used in this test case. The "no damage limit" is specified to be $0.3g$. The fragility model is shown in Figure 4.5.

The ground motion values at the location of the single asset are $[1.3, 0.044, 0.22, 1.0, 1.2]g$. The calculation of the damage state exceedance probabilities proceeds in exactly the same manner as demonstrated in Case 1a, except that for the ground motion values of $0.044g$ and $0.22g$, the probabilities for all four damage states are assumed to be zero.

GMF #	Site	PGA (g)
1	1	1.300
2	1	0.044
3	1	0.220
4	1	1.000
5	1	1.200

Table 4.44 – *Five precomputed ground motion fields at a single site*

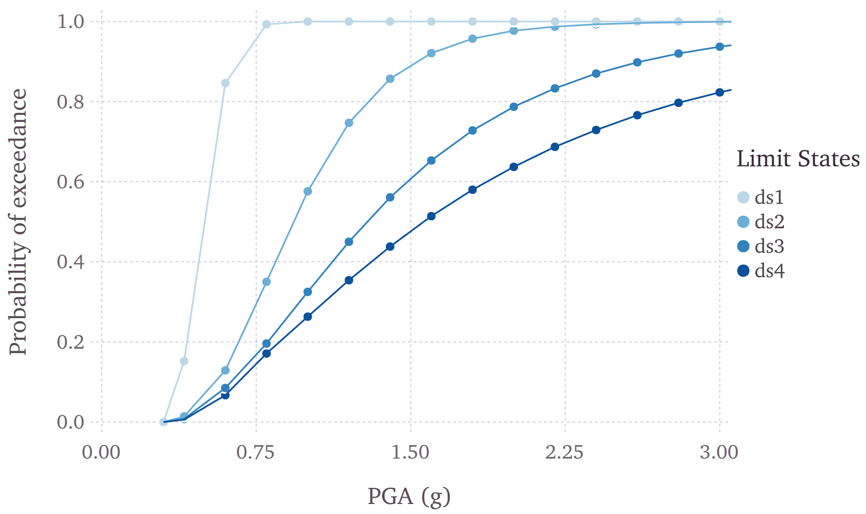

Figure 4.5 – *Discrete fragility model with four damage states and a nonzero no damage limit*

We have the following sets of damage state probabilities:

- GMF1: $[0.000, 0.198, 0.2965, 0.1095, 0.396]$
- GMF2: $[1.000, 0.000, 0.000, 0.000, 0.000]$
- GMF3: $[1.000, 0.000, 0.000, 0.000, 0.000]$
- GMF4: $[0.000, 0.424, 0.251, 0.062, 0.263]$
- GMF5: $[0.000, 0.253, 0.297, 0.096, 0.354]$

The mean and standard deviation of the four damage state probabilities and also the probability of observing no damage is now calculated using the above set of probabilities collected from each of the five ground motion simulations. Table 4.45 shows the comparison of the OpenQuake result with the expected result.

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	0.4000	0.4000	0.00%
		Std.	0.5477	0.5477	0.00%
	ds1	Mean	0.1750	0.1750	0.00%
		Std.	0.1802	0.1802	0.00%
	ds2	Mean	0.1689	0.1689	0.00%
		Std.	0.1553	0.1553	0.00%
	ds3	Mean	0.0535	0.0535	0.00%
		Std.	0.0518	0.0518	0.00%
	ds4	Mean	0.2026	0.2026	0.00%
		Std.	0.1911	0.1911	0.00%

Table 4.45 – *Results for scenario damage test case 1c*

4.2.1.4 Case 1d

Test Case 1d repeats the exercise from Case 1b, with the difference that the continuous fragility function specifies a minimum ground motion intensity, below which the probability of exceedance for all damage states is assumed to be zero.

Table 4.44 lists the five ground motion values used in this test case, and Table 4.42 the continuous fragility function used in this test case. The "minimum intensity level" is specified to be $0.3g$.

The ground motion values at the location of the single asset are $[1.3, 0.044, 0.22, 1.0, 1.2]g$. The calculation of the damage state exceedance probabilities proceeds in exactly the same manner as demonstrated in Case 1a, except that for the ground motion values of $0.044g$ and $0.22g$, the probabilities for all four damage states are assumed to be zero.

We have the following sets of damage state probabilities:

- GMF1: $[0.0436, 0.191, 0.207, 0.162, 0.397]$
- GMF2: $[1.000, 0.000, 0.000, 0.000, 0.000]$
- GMF3: $[1.000, 0.000, 0.000, 0.000, 0.000]$
- GMF4: $[0.091, 0.272, 0.226, 0.148, 0.263]$
- GMF5: $[0.055, 0.215, 0.215, 0.160, 0.354]$

The mean and standard deviation of the four damage state probabilities and also the probability of observing no damage is now calculated using the above set of probabilities collected from each of the five ground motion simulations. Table 4.46 shows the comparison of the OpenQuake

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	0.4379	0.4379	0.00%
		Std.	0.5134	0.5134	0.00%
	ds1	Mean	0.1356	0.1356	0.00%
		Std.	0.1272	0.1272	0.00%
	ds2	Mean	0.1296	0.1296	0.00%
		Std.	0.1185	0.1185	0.00%
	ds3	Mean	0.0940	0.0940	0.00%
		Std.	0.0860	0.0860	0.00%
	ds4	Mean	0.2028	0.2028	0.00%
		Std.	0.1913	0.1913	0.00%

Table 4.46 – *Results for scenario damage test case 1d*

result with the expected result.

4.2.1.5 Case 1e

The purpose of this case is to test the computation of the mean and standard deviation of the damage state probabilities when the ground motion fields are not predefined as in previous cases, but generated by OpenQuake by sampling from the distribution defined by the selected ground motion prediction equation. Ten thousand ground motion fields are generated for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008).

Table 4.13 lists five of the ten thousand ground motion values generated by OpenQuake. Table 4.40 shows the set of ground motion intensity levels and corresponding probabilities of exceedance for the four damage states for the discrete fragility function used in this test case.

In order to check the OpenQuake results, an alternate implementation of the calculator algorithm in the programming language Julia is used for comparison. In order to provide a representative baseline for the comparison, one million ground motion fields are used in the Julia calculation.

The mean and standard deviation of the logarithm of the ground motion calculated at the location of the asset as obtained by using the Boore and Atkinson (2008) equation are -0.648 and 0.564 respectively. Assuming a lognormal distribution for the variability in the ground motion, one million ground motion values are generated using Julia with these logarithmic mean and standard deviation values.

For each simulated ground motion value, the damage state exceedance probabilities are obtained through interpolation on the discrete fragility function specified for this case. The probabilities of damage state occurrence are calculated by the pairwise differences of the exceedance probabilities of adjacent damage states as described in Case 1a.

The mean and standard deviation of the damage state probabilities and also the probability of observing no damage are finally calculated using the above sets of probabilities collected from each of the one million ground motion simulations. Table 4.47 shows the comparison of the

Asset	Damage State	Result	Julia	OpenQuake	Difference
a1	none	Mean	0.4550	0.4575	-0.53%
		Std.	0.3934	0.3931	0.07%
	ds1	Mean	0.3400	0.3401	-0.02%
		Std.	0.2512	0.2520	-0.31%
	ds2	Mean	0.0775	0.0767	0.97%
		Std.	0.0949	0.0945	0.41%
	ds3	Mean	0.0240	0.0238	0.94%
		Std.	0.0343	0.0342	0.16%
	ds4	Mean	0.1036	0.1020	1.47%
		Std.	0.1405	0.1384	1.50%

Table 4.47 – *Results for scenario damage test case 1e*

OpenQuake result with the expected result.

4.2.1.6 Case 1f

This test case is identical to Case 1e described above, except for the use of a continous fragility function (see Table 4.42) instead of the discrete fragility function used in the previous case. Table 4.48 shows the comparison of the OpenQuake result with the expected result.

Asset	Damage State	Result	Julia	OpenQuake	Difference
a1	none	Mean	0.3729	0.3736	-0.19%
		Std.	0.2419	0.2403	0.66%
	ds1	Mean	0.2988	0.2998	-0.31%
		Std.	0.0837	0.0824	1.56%
	ds2	Mean	0.1427	0.1426	0.05%
		Std.	0.0676	0.0673	0.43%
	ds3	Mean	0.0731	0.0728	0.32%
		Std.	0.0504	0.0503	0.26%
	ds4	Mean	0.1125	0.1112	1.19%
		Std.	0.1367	0.1345	1.61%

Table 4.48 – *Results for scenario damage test case 1f*

4.2.1.7 Case 1g

Case 1g is designed to test a simple multiplication of the mean and standard deviation of damage state probabilities by the number of units comprising an asset. When an asset comprises more than one unit, the Scenario Damage calculator returns the expected fraction of buildings in each damage state, and the corresponding standard deviation. This case is thus designed identical to Case 1f, except that the single asset used in this case comprises three units, instead of one as in the previous case. The expected results should be three times those obtained in Case 1f. Table

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	1.1187	1.1209	-0.19%
		Std.	0.7258	0.7210	0.66%
	ds1	Mean	0.8965	0.8993	-0.31%
		Std.	0.2511	0.2472	1.56%
	ds2	Mean	0.4281	0.4279	0.05%
		Std.	0.2028	0.2020	0.43%
	ds3	Mean	0.2192	0.2185	0.32%
		Std.	0.1513	0.1509	0.26%
	ds4	Mean	0.3375	0.3335	1.19%
		Std.	0.4102	0.4036	1.61%

Table 4.49 – *Results for scenario damage test case 1g*

4.49 shows the comparison of the OpenQuake result with the expected result.

4.2.2 Multiple asset tests

The multiple asset test cases are designed to test the damage state aggregation functions of the scenario damage calculator, such as:

- damage distribution per taxonomy
- damage distribution for the portfolio

4.2.2.1 Case 2a

The purpose of this case is to test the basic elements of a scenario damage calculation involving multiple assets, such as the computation of the mean and standard deviation of the number of buildings in each damage state for assets aggregated by taxonomy, and also for the overall portfolio comprising all assets.

The list of assets in the exposure model used in this case is given in Table 4.29.

Five precomputed ground motion fields are used as the starting point for this case. These ground motion fields take into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied. Table 4.30 lists the ground motion fields used in this test case.

Table 4.50 shows the parameters for the continuous lognormal fragility functions for the three taxonomies.

Consider asset $a3$, which has the taxonomy $tax1$. The ground motion values at the location of the single asset are $[0.15, 0.05, 0.05, 0.15, 0.15]g$. Consider the first value of $PGA = 0.15g$. The exceedance probabilities for the four damage states ds_1, ds_2, ds_3, ds_4 are obtained by employing the equation for the cumulative distribution function for the lognormal distribution, using the

Taxonomy	LS	Mean IML	Std. IML
tax1	**ds1**	0.50	0.40
	ds2	1.00	0.80
	ds3	1.50	1.20
	ds4	2.00	1.60
tax2	**ds1**	1.00	0.80
	ds2	1.50	1.20
	ds3	2.50	2.00
	ds4	4.00	3.20
tax3	**ds1**	1.20	0.90
	ds2	1.80	1.50
	ds3	3.00	2.00
	ds4	5.00	3.50

Table 4.50 – *Fragility functions for three taxonomies*

parameters specified by the fragility function for $tax1$. This process is described in Case 1b, and is not repeated here. The damage state probabilities are multiplied by the number of units comprising asset $a3$, which in this case is one. This gives us the expected number of buildings for asset $a3$ in each damage state for the five ground motions:

- GMF1: $[0.9131, 0.0774, 0.0077, 0.0013, 0.0004]$
- GMF2: $[0.9982, 0.0018, 0.0000, 0.0000, 0.0000]$
- GMF3: $[0.9982, 0.0018, 0.0000, 0.0000, 0.0000]$
- GMF4: $[0.9131, 0.0774, 0.0077, 0.0013, 0.0004]$
- GMF5: $[0.9131, 0.0774, 0.0077, 0.0013, 0.0004]$

The mean number of buildings with no damage for asset $a3$ is thus calculated as $(0.9131 + 0.9982 + 0.9982 + 0.9131 + 0.9131)/5 = 0.94716$. The standard deviation of the number of buildings with no damage for asset $a3$ is 0.04661. Similarly, the mean and standard deviation of the number of buildings with no damage for the other assets are calculated and listed below:

- $a1$ (tax1): mean = 0.28370; std.dev = 0.29192
- $a2$ (tax2): mean = 0.89301; std.dev = 0.11739
- $a3$ (tax1): mean = 0.94716; std.dev = 0.04661
- $a4$ (tax3): mean = 0.61303; std.dev = 0.24347
- $a5$ (tax1): mean = 0.59337; std.dev = 0.41368
- $a6$ (tax2): mean = 0.77733; std.dev = 0.18346
- $a7$ (tax1): mean = 0.65092; std.dev = 0.24269

Repeating this for each of the other damage states, we can also compute the mean and standard deviation of the number of buildings in each damage state for the seven assets. Table 4.51 shows the comparison of the OpenQuake result with the expected result for assets $a1$, $a2$, and $a3$.

Now, aggregating the expected number of buildings of taxonomy $tax1$ with no damage for

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	0.2837	0.2837	0.00%
		Std.	0.2919	0.2919	0.00%
	ds1	Mean	0.2625	0.2625	0.00%
		Std.	0.1002	0.1002	0.00%
	ds2	Mean	0.1568	0.1568	0.00%
		Std.	0.0767	0.0767	0.00%
	ds3	Mean	0.0962	0.0962	0.00%
		Std.	0.0629	0.0629	0.00%
	ds4	Mean	0.2008	0.2008	0.00%
		Std.	0.2159	0.2159	0.00%
a2	none	Mean	0.8930	0.8930	0.00%
		Std.	0.1174	0.1174	0.00%
	ds1	Mean	0.0653	0.0653	0.00%
		Std.	0.0666	0.0666	0.00%
	ds2	Mean	0.0328	0.0328	0.00%
		Std.	0.0392	0.0392	0.00%
	ds3	Mean	0.0074	0.0074	0.00%
		Std.	0.0096	0.0096	0.00%
	ds4	Mean	0.0014	0.0014	0.00%
		Std.	0.0019	0.0019	0.00%
a3	none	Mean	0.9472	0.9472	0.00%
		Std.	0.0466	0.0466	0.00%
	ds1	Mean	0.0471	0.0471	0.00%
		Std.	0.0415	0.0415	0.00%
	ds2	Mean	0.0047	0.0047	0.00%
		Std.	0.0042	0.0042	0.00%
	ds3	Mean	0.0008	0.0008	0.00%
		Std.	0.0007	0.0007	0.00%
	ds4	Mean	0.0003	0.0003	0.00%
		Std.	0.0002	0.0002	0.00%

Table 4.51 – *Results for scenario damage test case 2a — individual assets*

each of the five ground motions, we have the following for $tax1$: $0.28370+0.94716+0.59337+0.65092 = 2.47515$. Repeating this for each of the other damage states, we can also compute the expected total number of buildings in each damage state for assets of taxonomy $tax1$. Table 4.52 shows the comparison of the OpenQuake result with the expected result for taxonomies $tax1$, $tax2$, and $tax3$.

Taxonomy	Damage State	Result	Expected	OpenQuake	Difference
tax1	none	Mean	2.4752	2.4752	0.00%
	ds1	Mean	0.7294	0.7294	0.00%
	ds2	Mean	0.3257	0.3257	0.00%
	ds3	Mean	0.1736	0.1736	0.00%
	ds4	Mean	0.2962	0.2962	0.00%
tax2	none	Mean	1.6703	1.6703	0.00%
	ds1	Mean	0.1832	0.1832	0.00%
	ds2	Mean	0.1082	0.1082	0.00%
	ds3	Mean	0.0304	0.0304	0.00%
	ds4	Mean	0.0078	0.0078	0.00%
tax3	none	Mean	0.6130	0.6130	0.00%
	ds1	Mean	0.1422	0.1422	0.00%
	ds2	Mean	0.1800	0.1800	0.00%
	ds3	Mean	0.0467	0.0467	0.00%
	ds4	Mean	0.0181	0.0181	0.00%

Table 4.52 – *Results for scenario damage test case 2a — aggregated by taxonomy*

Finally, aggregating the expected number of buildings with no damage for across all taxonomies for each of the five ground motions, we have the following for the overall portfolio: $0.28370+0.89301+0.94716+0.61303+0.59337+0.77733+0.65092 = 4.75852$. Repeating this for each of the other damage states, we can also compute the expected total number of buildings in each damage state for the overall portfolio. Table 4.53 shows the comparison of the OpenQuake result with the expected result for the overall portfolio.

Damage State	Result	Expected	OpenQuake	Difference
none	Mean	4.7585	4.7585	0.00%
ds1	Mean	1.0547	1.0547	0.00%
ds2	Mean	0.6140	0.6140	0.00%
ds3	Mean	0.2507	0.2507	0.00%
ds4	Mean	0.3221	0.3221	0.00%

Table 4.53 – *Results for scenario damage test case 2a — overall portfolio*

4.2.2.2 Case 2b

The purpose of this case is to test the computation of the mean and standard deviation of the damage state probabilities for a portfolio of assets when the ground motion fields are not predefined, but generated by OpenQuake by sampling from the distribution defined by the selected ground motion prediction equation.

The exposure and fragility models used in this case are the same as those used in Case 2a. Ten thousand ground motion fields are generated for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied.

Table 4.33 lists five of the ten thousand ground motion fields generated by the OpenQuake scenario hazard calculator.

In order to provide a representative baseline for the comparison, one million ground motion fields are used in the Julia implementation of the calculator.

Asset	Damage State	Result	Expected	OpenQuake	Difference
a1	none	Mean	0.3726	0.3726	0.00%
		Std.	0.2420	0.2405	0.61%
	ds1	Mean	0.2988	0.2996	-0.26%
		Std.	0.0837	0.0824	1.65%
	ds2	Mean	0.1428	0.1429	-0.08%
		Std.	0.0676	0.0673	0.53%
	ds3	Mean	0.0731	0.0731	0.02%
		Std.	0.0505	0.0505	-0.09%
	ds4	Mean	0.1127	0.1118	0.78%
		Std.	0.1368	0.1345	1.70%

Table 4.54 – *Results for scenario damage test case 2b — individual assets*

Table 4.51 shows the comparison of the OpenQuake result with the expected result for assets $a1$, $a2$, and $a3$. Table 4.52 shows the comparison of the OpenQuake result with the expected result for taxonomies $tax1$, $tax2$, and $tax3$. Finally, Table 4.56 shows the comparison of the OpenQuake result with the expected result for the overall portfolio.

4.2.3 Calculation with logic-trees

4.2.3.1 Case 3a

The OpenQuake scenario damage calculator allows the user to employ more than one ground motion prediction equation (GMPE) for computing the ground motion fields used for the loss calculation. The mean and standard deviation of the damage state probabilities (or fractions), are calculated and output for each GMPE branch independently. No sampling is involved, and any branch weights assigned to the different GMPE branches are ignored.

A single asset is used in this test case. Table 4.42 shows the parameters of the continuous

Taxonomy	Damage State	Result	Expected	OpenQuake	Difference
tax1	none	Mean	2.4752	2.4752	0.00%
	ds1	Mean	0.7294	0.7294	0.00%
	ds2	Mean	0.3257	0.3257	0.00%
	ds3	Mean	0.1736	0.1736	0.00%
	ds4	Mean	0.2962	0.2962	0.00%
tax2	none	Mean	1.6703	1.6703	0.00%
	ds1	Mean	0.1832	0.1832	0.00%
	ds2	Mean	0.1082	0.1082	0.00%
	ds3	Mean	0.0304	0.0304	0.00%
	ds4	Mean	0.0078	0.0078	0.00%
tax3	none	Mean	0.6130	0.6130	0.00%
	ds1	Mean	0.1422	0.1422	0.00%
	ds2	Mean	0.1800	0.1800	0.00%
	ds3	Mean	0.0467	0.0467	0.00%
	ds4	Mean	0.0181	0.0181	0.00%

Table 4.55 – *Results for scenario damage test case 2a — aggregated by taxonomy*

Damage State	Result	Expected	OpenQuake	Difference
none	Mean	4.7585	4.7585	0.00%
ds1	Mean	1.0547	1.0547	0.00%
ds2	Mean	0.6140	0.6140	0.00%
ds3	Mean	0.2507	0.2507	0.00%
ds4	Mean	0.3221	0.3221	0.00%

Table 4.56 – *Results for scenario damage test case 2a — overall portfolio*

lognormal fragility function used in this test case.

The two ground motion prediction equations used are Boore and Atkinson (2008), and Chiou and Youngs (2008). Ten thousand ground motion fields are generated using OpenQuake for the given rupture, taking into consideration both the inter-event and intra-event variability in the ground motion, for both GMPE branches. The rest of the loss calculation procedure follows the same steps as described earlier in Case 1e.

For comparison, one million ground motion fields are generated using both GMPEs using Julia.

Branch	Asset	Damage State	Result	Julia	OpenQuake	Difference
BA2008	a1	none	Mean	0.3725	0.3736	-0.30%
			Std.	0.2420	0.2403	0.69%
		ds1	Mean	0.2988	0.2998	-0.31%
			Std.	0.0838	0.0824	1.64%
		ds2	Mean	0.1428	0.1426	0.15%
			Std.	0.0676	0.0673	0.44%
		ds3	Mean	0.0732	0.0728	0.47%
			Std.	0.0504	0.0503	0.27%
		ds4	Mean	0.1127	0.1112	1.33%
			Std.	0.1367	0.1345	1.56%
CY2008	a1	none	Mean	0.3086	0.3093	-0.25%
			Std.	0.2218	0.2197	0.92%
		ds1	Mean	0.3019	0.3032	-0.45%
			Std.	0.0814	0.0799	1.92%
		ds2	Mean	0.1590	0.1590	-0.02%
			Std.	0.0617	0.0613	0.61%
		ds3	Mean	0.0862	0.0859	0.34%
			Std.	0.0500	0.0498	0.40%
		ds4	Mean	0.1444	0.1426	1.30%
			Std.	0.1535	0.1512	1.53%

Table 4.57 – *Results for scenario damage test case 3a*

Table 4.57 shows the comparison of the OpenQuake results with the expected results.

4.3 Classical Risk Calculator

The tests for the classical PSHA-based risk calculator assume the correct computation of the hazard curves at the locations of the assets in the exposure model. Thus, the risk tests implicitly rely on the acceptance tests for the classical PSHA-based hazard calculator.

The source model used for the tests comprises a single vertical strike-slip fault with a Gutenberg-Richter b-value equal to 0.9 and a slip rate of 2 mm/yr. The MFD is a Gutenberg-Richter distribution truncated between magnitudes 5.0 and 6.5, while the Ground Motion Prediction Equation (GMPE) used is Boore and Atkinson (2008).

Details of the fault geometry are given below:

Fault type: Strike slip
Fault dip: 90°
Fault plane depths: 0–12 km
Fault coordinates:
South end: 38.0000°N, 122.0000°W
North end: 38.2248°N, 122.0000°W

Figure 4.6, shows the fault described above and the site geometry for the test cases described in the following sections. The single asset tests use only site 1 shown in the figure, whereas the multiple asset tests use all seven of the sites. The geometry of the fault and locations of the sites match those selected by Thomas et al. (2010) in their effort to verify PSHA computer programs.

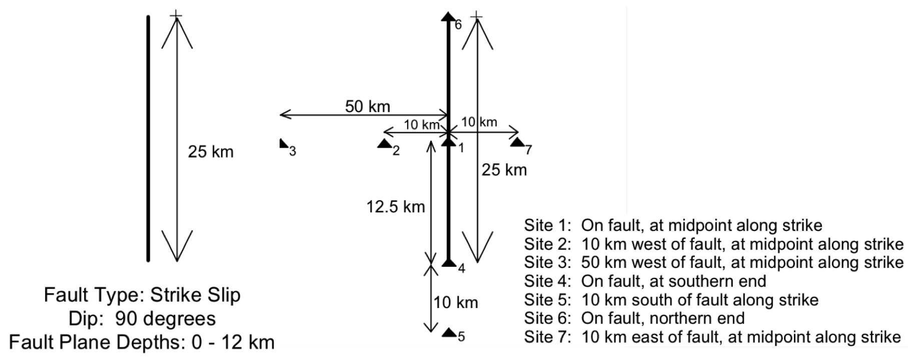

Figure 4.6 – *Fault and site geometry for the classical risk tests. From Thomas et al. (2010)*

The complete collection of input models and job configuration files used in these test cases can be accessed here: https://github.com/gem/oq-risklib/tree/master/openquake/qa_tests_data/classical_risk

4.3.1 Single asset tests

The single asset test cases are designed to test the basic elements of the classical-PSHA based risk calculator, such as:

- asset loss ratio exceedance curve computation
- asset loss exceedance curve computation

The location and taxonomy of the single asset in the exposure model used for the single-asset test cases for the classical risk calculator are given in Table 4.1.

4.3.1.1 Case 1a

Table 4.3 shows the mean loss ratios and corresponding coefficients of variation for the vulnerability function used in this test case.

When the exposure model and vulnerability model are provided to the OpenQuake classical PSHA-based hazard calculator, OpenQuake computes the hazard curves at the locations of the assets in the exposure model and at the specific intensity levels used in the vulnerability functions.

PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	1.40g	1.60g	1.80g	2.00g
P.O.E.	3.896×10^{-2}	2.222×10^{-2}	8.171×10^{-3}	3.070×10^{-3}	1.230×10^{-3}	5.195×10^{-4}	2.254×10^{-4}	9.918×10^{-5}	4.353×10^{-5}	1.830×10^{-5}	6.925×10^{-6}

Table 4.58 – *Hazard curve for PGA at a single site*

The intensity levels for the hazard curve are extracted from the vulnerability function: $[0.05, 0.20, 0.40, 0.60, 0.80, 1.00, 1.20, 1.40, 1.60, 1.80, 2.00]$. The hazard curve gives the probabilities of exceedance for a set of intensity levels within a specified time period. The time period in this case, t_H, is one year. The hazard curve at the location of the single asset used in this test case is shown in Table 4.58.

The probabilities of exceedance are: $[3.896 \times 10^{-2}, 2.222 \times 10^{-2}, 8.171 \times 10^{-3}, 3.070 \times 10^{-3}, 1.230 \times 10^{-3}, 5.195 \times 10^{-4}, 2.254 \times 10^{-4}, 9.918 \times 10^{-5}, 4.353 \times 10^{-5}, 1.830 \times 10^{-5}, 6.925 \times 10^{-6}]$. The probabilities of exceedance are first converted to annual rates (or frequencies) of exceedance by employing the Poissonion conversion:

$$\lambda(iml) = \frac{-\ln[1 - prob(IML > iml, t_H)]}{t_H} \tag{4.7}$$

The annual frequencies of exceedance are: $[3.974 \times 10^{2}, 2.247 \times 10^{2}, 8.205 \times 10^{3}, 3.075 \times 10^{3}, 1.231 \times 10^{3}, 5.197 \times 10^{4}, 2.254 \times 10^{4}, 9.918 \times 10^{5}, 4.353 \times 10^{5}, 1.829 \times 10^{5}, 6.925 \times 10^{6}]$.

The annual frequencies of occurrence are estimated by the differentiation of the annual frequencies of exceedance: $[1.727 \times 10^{2}, 1.426 \times 10^{2}, 5.130 \times 10^{3}, 1.845 \times 10^{3}, 7.109 \times 10^{4}, 2.942 \times 10^{4}, 1.262 \times 10^{4}, 5.565 \times 10^{5}, 2.524 \times 10^{5}, 1.137 \times 10^{5}]$.

The loss ratios at which the loss curve exceedance probabilities are calculated are obtained from the vulnerability function and the parameter 'steps_per_interval'. The default value of 'steps_per_interval' is one, which is the value used in this case. The loss ratios in the vulnerability function are $[0.01, 0.04, 0.10, 0.20, 0.33, 0.50, 0.67, 0.80, 0.90, 0.96, 0.99]$.

The vulnerability model is then transformed into a matrix describing probabilities of exceedance for the selected set of loss ratios conditional on the set of ground motion intensity levels. Since there is no variability in the loss ratio, calculation of the loss curves is straightforward in this case. Since the coefficients of variation in the vulnerability function are all zero, the lognormal distribution devolves into the degenerate distribution. The loss ratio exceedance matrix in this case is shown in Table 4.59.

Now, the sum product of each row of the conditional loss ratio exceedance matrix with the annual frequencies of occurrence of the respective intensity levels gives the annual frequency of exceedance for the respective loss ratios. The loss ratio annual frequencies of exceedance thus calculated are: $[3.973 \times 10^{2}, 3.110 \times 10^{2}, 1.533 \times 10^{2}, 5.633 \times 10^{3}, 2.146 \times 10^{3}, 8.682 \times 10^{4}, 3.656 \times 10^{4}, 1.554 \times 10^{4}, 6.443 \times 10^{5}, 2.399 \times 10^{5}]$.

The probabilities of exceedance of the set of loss ratios are obtained by converting the frequencies of exceedance $\lambda_{L \geq l}$ back into probabilities $prob(L \geq l, t_R)$ of exceedance over the

LR ∣ PGA	**0.05g**	**0.20g**	**0.40g**	**0.60g**	**0.80g**	**1.00g**	**1.20g**	**...**	**2.00g**
0.01	1	1	1	1	1	1	1	...	1
0.04	0	1	1	1	1	1	1	...	1
0.10	0	0	1	1	1	1	1	...	1
0.20	0	0	0	1	1	1	1	...	1
0.33	0	0	0	0	1	1	1	...	1
0.50	0	0	0	0	0	1	1	...	1
0.67	0	0	0	0	0	0	1	...	1
0.80	0	0	0	0	0	0	0	...	1
0.90	0	0	0	0	0	0	0	...	1
0.96	0	0	0	0	0	0	0	...	1
0.99	0	0	0	0	0	0	0	...	1
1.00	0	0	0	0	0	0	0	...	0

Table 4.59 – *Conditional loss ratio exceedance matrix for classical risk test case 1a*

exposure time period t_R by using the Poissonion assumption:

$$prob(L \geq l, t_R) = 1 - \exp(-\lambda_{L \geq l} \times t_R) \tag{4.8}$$

The loss curve probabilities of exceedance thus calculated are: $[3.895 \times 10^2, 3.062 \times 10^2, 1.521 \times 10^2, 5.617 \times 10^3, 2.144 \times 10^3, 8.678 \times 10^4, 3.655 \times 10^4, 1.554 \times 10^4, 6.443 \times 10^5, 2.399 \times 10^5, 5.683 \times 10^6]$.

The loss curve thus calculated above is compared with the loss curve obtained using the OpenQuake classical PSHA based risk calculator in Figure 4.7.

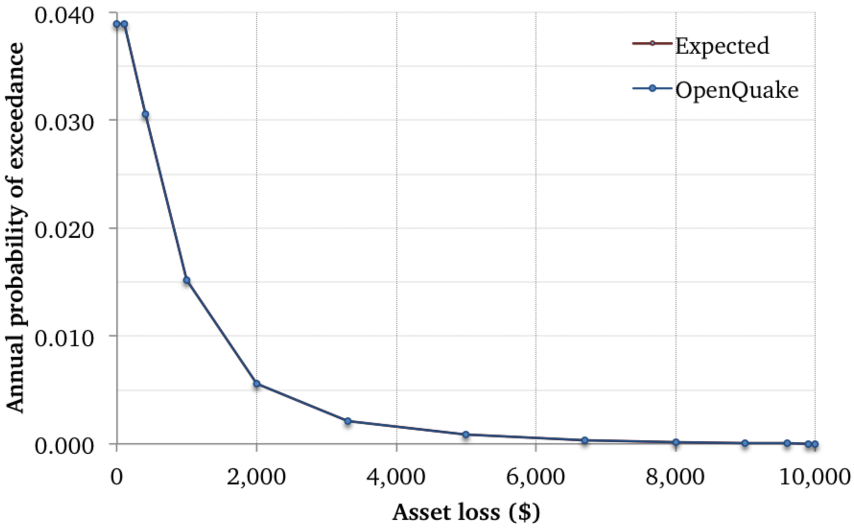

Figure 4.7 – *Loss curve comparison for classical risk test case 1a*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.60

Result	Expected	OpenQuake	Difference
Average loss	47.63	47.63	0.00%

Table 4.60 – *Results for classical risk test case 1a*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.3.1.2 Case 1b

This test case is identical to Case 1a described above, except for the use of the Beta distribution for the vulnerability functions instead of the lognormal distribution. Since the coefficients of variation in the vulnerability function are all zero, once again the Beta distribution devolves into the degenerate distribution as in the previous case. The results for this test case should be exactly the same as in Case 1a. Table 4.61 shows the comparison of the OpenQuake result for average annual loss with the expected result.

Result	Expected	OpenQuake	Difference
Average loss	47.63	47.63	0.00%

Table 4.61 – *Results for classical risk test case 1b*

4.3.1.3 Case 1c

This test case repeats the exercise from Case 1a using a vulnerability model with nonzero coefficients of variation. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation for the vulnerability function used in this test case.

Apart from the computation of the conditional loss ratio exceedance matrix, the steps for computing the loss exceedance curve in this case are the same as those employed in Case 1a. The conditional loss ratio exceedance matrix in this case is populated by evaluating the complementary cumulative distribution function (CCDF) of the lognormal distribution at each of the prescribed intensity levels, for the set of loss ratios.

The loss ratio exceedance matrix in this case is shown in Table 4.62.

The loss curve thus calculated above is compared with the loss curve obtained using the OpenQuake classical PSHA based risk calculator in Figure 4.8.

The area under the annual loss exceedance curve gives the average annual loss. Table 4.63 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.3.1.4 Case 1d

This test case is identical to Case 1c described above, except for the use of the Beta distribution for the vulnerability functions instead of the lognormal distribution. The conditional loss ratio exceedance matrix in this case is populated by evaluating the complementary cumulative distribution function (CCDF) of the Beta distribution at each of the prescribed intensity levels, for the set of loss ratios.

The loss ratio exceedance matrix in this case is shown in Table 4.64.

LR | PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
0.01	0.494	1.000	1.000	1.000	1.000	1.000	1.000	...	1.000
0.04	0.000	0.476	1.000	1.000	1.000	1.000	1.000	...	1.000
0.10	0.000	0.000	0.453	0.980	0.999	1.000	1.000	...	1.000
0.20	0.000	0.000	0.001	0.438	0.881	0.986	0.999	...	1.000
0.33	0.000	0.000	0.000	0.039	0.427	0.812	0.959	...	1.000
0.50	0.000	0.000	0.000	0.001	0.094	0.424	0.730	...	1.000
0.67	0.000	0.000	0.000	0.000	0.017	0.170	0.427	...	1.000
0.80	0.000	0.000	0.000	0.000	0.005	0.079	0.253	...	1.000
0.90	0.000	0.000	0.000	0.000	0.002	0.043	0.162	...	0.999
0.96	0.000	0.000	0.000	0.000	0.001	0.030	0.122	...	0.844
0.99	0.000	0.000	0.000	0.000	0.001	0.025	0.106	...	0.494
1.00	0.000	0.000	0.000	0.000	0.001	0.023	0.101	...	0.363

Table 4.62 – *Conditional loss ratio exceedance matrix for classical risk test case 1c*

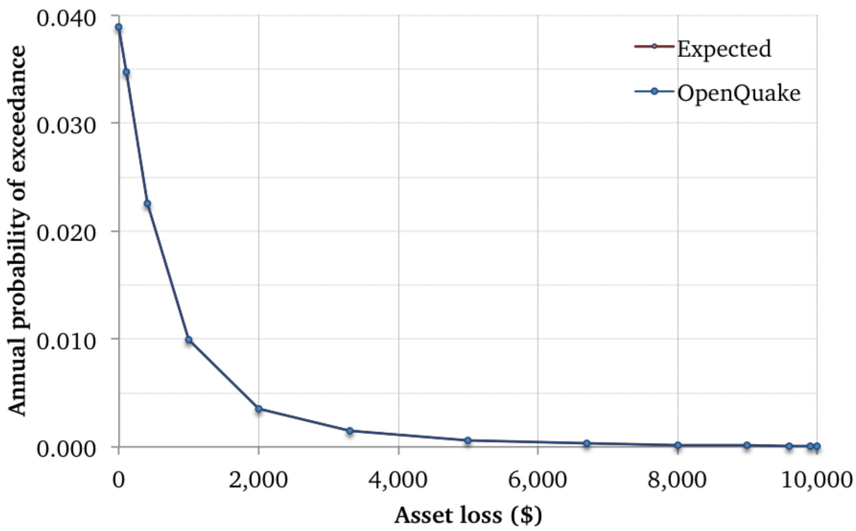

Figure 4.8 – *Loss curve comparison for classical risk test case 1c*

Result	Expected	OpenQuake	Difference
Average loss	35.13	35.13	0.00%

Table 4.63 – *Results for classical risk test case 1c*

LR | PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
0.01	0.496	1.000	1.000	1.000	1.000	1.000	1.000	...	1.000
0.04	0.000	0.485	0.999	1.000	1.000	0.999	0.996	...	1.000
0.10	0.000	0.000	0.472	0.959	0.984	0.987	0.982	...	1.000
0.20	0.000	0.000	0.000	0.468	0.844	0.928	0.944	...	1.000
0.33	0.000	0.000	0.000	0.032	0.473	0.778	0.871	...	1.000
0.50	0.000	0.000	0.000	0.000	0.100	0.500	0.738	...	1.000
0.67	0.000	0.000	0.000	0.000	0.006	0.222	0.563	...	0.999
0.80	0.000	0.000	0.000	0.000	0.000	0.072	0.394	...	0.995
0.90	0.000	0.000	0.000	0.000	0.000	0.013	0.234	...	0.976
0.96	0.000	0.000	0.000	0.000	0.000	0.001	0.115	...	0.925
0.99	0.000	0.000	0.000	0.000	0.000	0.000	0.038	...	0.822
1.00	0.000	0.000	0.000	0.000	0.000	0.000	0.000	...	0.000

Table 4.64 – *Conditional loss ratio exceedance matrix for classical risk test case 1d*

The loss curve thus calculated above is compared with the loss curve obtained using the OpenQuake classical PSHA based risk calculator in Figure 4.9.

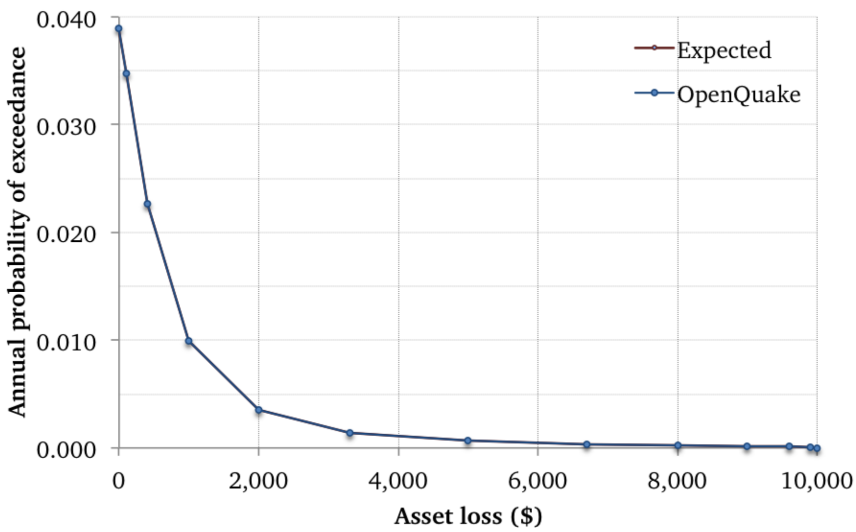

Figure 4.9 – *Loss curve comparison for classical risk test case 1d*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.65 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.3.1.5 Case 1e

This test case repeats the exercise from Case 1c using a vulnerability model with nonzero coefficients of variation, and using four 'steps_per_interval'. Each interval between the loss ratios specified in the vulnerability model is further divided into four equal subdivisions, thus ensuring a greater number of loss values at which the exceedance curve will be computed. For

Result	Expected	OpenQuake	Difference
Average loss	35.45	35.45	0.00%

Table 4.65 – *Results for classical risk test case 1d*

instance, the interval between the loss ratios $[0.10, 0.20]$ is now subdivided into the following loss ratios: $[0.100, 0.125, 0.150, 0.175, 0.200]$.

The loss curve thus calculated above is compared with the loss curve obtained using the OpenQuake classical PSHA based risk calculator in Figure 4.10.

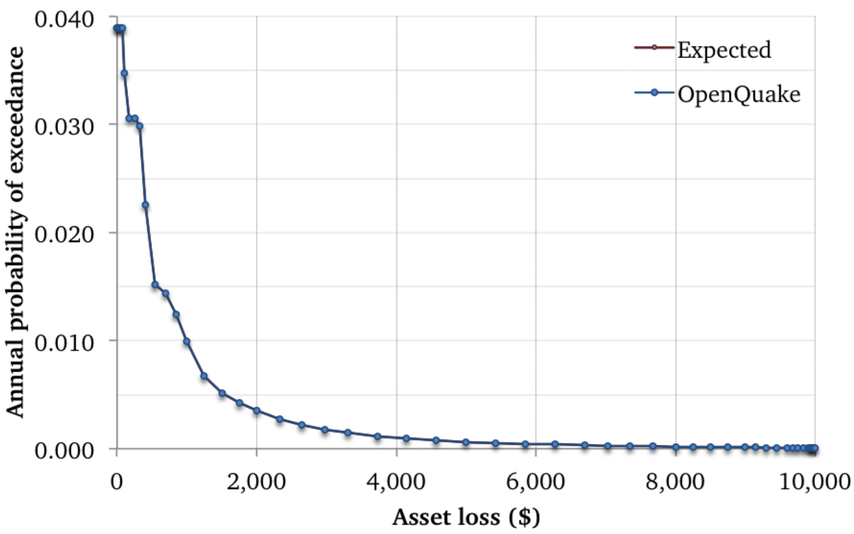

Figure 4.10 – *Loss curve comparison for classical risk test case 1e*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.66

Result	Expected	OpenQuake	Difference
Average loss	33.25	33.25	0.00%

Table 4.66 – *Results for classical risk test case 1e*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.3.1.6 Case 2a

In addition to computing direct structural losses, OpenQuake also provides support for computing losses incurred for the following other loss types:

- Non-structural losses
- Contents losses
- Downtime, or business interruption losses
- Occupant fatalities

The purpose of this case is to test the calculation of the loss exceedance curve and average annual loss for the non-structural components of an asset. The replacement value of the non-structural components for the asset used in this case is $15,000$. Table 4.17 shows the mean loss ratios and corresponding coefficients of variation for the non-structural components vulnerability model used in this test case. Table 4.61 shows the comparison of the OpenQuake result for

Result	Expected	OpenQuake	Difference
Average loss	63.48	63.48	0.00%

Table 4.67 – *Results for classical risk test case 2a*

average annual nonstructural loss with the expected result.

4.3.1.7 Case 2b

The purpose of this case is to test the calculation of loss exceedance curve and average annual loss for the contents of an asset. The replacement value of the contents for the asset used in this case is $5,000$. Table 4.19 shows the mean loss ratios and corresponding coefficients of variation in the contents vulnerability function used in this test case. Table 4.68 shows the comparison of

Result	Expected	OpenQuake	Difference
Average loss	49.08	49.08	0.00%

Table 4.68 – *Results for classical risk test case 2b*

the OpenQuake result for average annual contents loss with the expected result.

4.3.1.8 Case 2c

The purpose of this case is to test the calculation of loss exceedance curve and average annual loss for the downtime, or business-interruption losses for an asset. The loss due to downtime, or business-interruption for the asset used in this case is $2,000/month$. Downtime losses are usually specified per unit time the asset will be unavailable for occupancy or use. Table 4.21 shows the mean loss ratios and corresponding coefficients of variation for the downtime vulnerability function used in this test case. Table 4.69 shows the comparison of the OpenQuake result for

Result	Expected	OpenQuake	Difference
Average loss	7.02	7.02	0.00%

Table 4.69 – *Results for classical risk test case 2c*

average annual downtime loss with the expected result.

4.3.1.9 Case 2d

The purpose of this case is to test the calculation of the exceedance curve for fatalities and the average annual occupant fatalities for an asset. The number of occupants for the asset used in this case are 2 (day), 4 (transit), and 6 (night). An average value of 4 occupants is used for the calculation of the exceedance curve and average annual fatalities. Table 4.21 shows the mean loss ratios and corresponding coefficients of variation for the occupants fatality vulnerability function used in this test case. Table 4.70 shows the comparison of the OpenQuake result for

Result	Expected	OpenQuake	Difference
Average annual fatalities	2.91×10^{-4}	2.91×10^{-4}	0.00%

Table 4.70 – *Results for classical risk test case 2d*

average annual fatalities with the expected result.

4.3.1.10 Case 3a

The purpose of this case is to test the computation of the loss exceedance probabilities when the time period associated with the loss curve calculation is different from the time period associated with the hazard curve calculation. In this case, the hazard curve is calculated for a time period of 50 years, and the loss curve is calculated for a time period of 75 years.

Table 4.7 shows the mean loss ratios and corresponding coefficients of variation for the vulnerability function used in this test case.

PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	1.40g	1.60g	1.80g	2.00g
P.O.E.	8.643×10^{-1}	7.171×10^{-1}	4.371×10^{-1}	2.364×10^{-1}	1.234×10^{-1}	6.427×10^{-2}	3.382×10^{-2}	1.802×10^{-2}	9.676×10^{-3}	5.192×10^{-3}	2.748×10^{-3}

Table 4.71 – *50-year hazard curve for PGA at a single site*

The intensity levels for the hazard curve are extracted from the vulnerability function: $[0.05, 0.20, 0.40, 0.60, 0.80, 1.00, 1.20, 1.40, 1.60, 1.80, 2.00]$. The hazard curve gives the probabilities of exceedance for a set of intensity levels within a specified time period. The time period in this case, t_H, is fifty years. The hazard curve at the location of the single asset used in this test case is shown in Table 4.71.

The probabilities of exceedance are: $[8.643 \times 10^{-1}, 7.171 \times 10^{-1}, 4.371 \times 10^{-1}, 2.364 \times 10^{-1}, 1.234 \times 10^{-1}, 6.427 \times 10^{-2}, 3.382 \times 10^{-2}, 1.802 \times 10^{-2}, 9.676 \times 10^{-3}, 5.192 \times 10^{-3}, 2.748 \times 10^{-3}]$. The probabilities of exceedance are first converted to annual rates (or frequencies) of exceedance by employing the Poissonion conversion:

$$\lambda(iml) = \frac{-\ln[1 - prob(IML > iml, t_H)]}{t_H} \tag{4.9}$$

The annual frequencies of exceedance are: $[3.994 \times 10^{-2}, 2.525 \times 10^{-2}, 1.149 \times 10^{-2}, 5.394 \times 10^{-3}, 2.633 \times 10^{-3}, 1.329 \times 10^{-3}, 6.881 \times 10^{-4}, 3.636 \times 10^{-4}, 1.945 \times 10^{-4}, 1.041 \times 10^{-4}, 5.504 \times 10^{-5}]$.

The annual frequencies of occurrence are estimated by the differentiation of the annual frequencies of exceedance: $[1.469 \times 10^{-2}, 1.376 \times 10^{-2}, 6.101 \times 10^{-3}, 2.760 \times 10^{-3}, 1.305 \times 10^{-3}, 6.404 \times 10^{-4}, 3.245 \times 10^{-4}, 1.692 \times 10^{-4}, 9.035 \times 10^{-5}, 4.907 \times 10^{-5}]$.

The loss ratios at which the loss curve exceedance probabilities are calculated are obtained from the vulnerability function and the parameter 'steps_per_interval'. The default value of 'steps_per_interval' is one, which is the value used in this case. The loss ratios in the vulnerability function are $[0.01, 0.04, 0.10, 0.20, 0.33, 0.50, 0.67, 0.80, 0.90, 0.96, 0.99]$.

The vulnerability model is then transformed into a matrix describing probabilities of exceedance for the selected set of loss ratios conditional on the set of ground motion intensity levels.

Since there is no variability in the loss ratio, calculation of the loss curves is straightforward in this case. Since the coefficients of variation in the vulnerability function are all zero, the lognormal distribution devolves into the degenerate distribution. The loss ratio exceedance matrix in this case is shown in Table 4.72.

LR | PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	...	2.00g
0.01	0.494	1.000	1.000	1.000	1.000	1.000	1.000	...	1.000
0.04	0.000	0.476	1.000	1.000	1.000	1.000	1.000	...	1.000
0.10	0.000	0.000	0.453	0.980	0.999	1.000	1.000	...	1.000
0.20	0.000	0.000	0.001	0.438	0.881	0.986	0.999	...	1.000
0.33	0.000	0.000	0.000	0.039	0.427	0.812	0.959	...	1.000
0.50	0.000	0.000	0.000	0.001	0.094	0.424	0.730	...	1.000
0.67	0.000	0.000	0.000	0.000	0.017	0.170	0.427	...	1.000
0.80	0.000	0.000	0.000	0.000	0.005	0.079	0.253	...	1.000
0.90	0.000	0.000	0.000	0.000	0.002	0.043	0.162	...	0.999
0.96	0.000	0.000	0.000	0.000	0.001	0.030	0.122	...	0.844
0.99	0.000	0.000	0.000	0.000	0.001	0.025	0.106	...	0.494
1.00	0.000	0.000	0.000	0.000	0.001	0.023	0.101	...	0.363

Table 4.72 – *Conditional loss ratio exceedance matrix for classical risk test case 3a*

Now, the sum product of each row of the conditional loss ratio exceedance matrix with the annual frequencies of occurrence of the respective intensity levels gives the annual frequency of exceedance for the respective loss ratios. The loss ratio annual frequencies of exceedance thus calculated are: $[3.988\times10^{-2}, 3.617\times10^{-2}, 2.509\times10^{-2}, 1.280\times10^{-2}, 5.654\times10^{-3}, 2.765\times10^{-3}, 1.408\times10^{-3}, 7.772\times10^{-4}, 4.896\times10^{-4}, 3.276\times10^{-4}, 2.457\times10^{-4}, 2.037\times10^{-4}, 1.903\times10^{-4}]$.

The probabilities of exceedance of the set of loss ratios are obtained by converting the annual frequencies of exceedance back into probabilities of exceedance over 75 years by using the Poissonion equation. The loss curve probabilities of exceedance for a time period of 75 years are: $[9.498\times10^{-1}, 9.336\times10^{-1}, 8.477\times10^{-1}, 6.170\times10^{-1}, 3.456\times10^{-1}, 1.873\times10^{-1}, 1.002\times10^{-1}, 5.663\times10^{-2}, 3.606\times10^{-2}, 2.427\times10^{-2}, 1.826\times10^{-2}, 1.516\times10^{-2}, 1.417\times10^{-2}]$.

The loss curve thus calculated above is compared with the loss curve obtained using the OpenQuake classical PSHA based risk calculator in Figure 4.11.

The area under the loss exceedance curve gives the expected loss over 75 years. Table 4.73

Result	Expected	OpenQuake	Difference
Average loss	2,115.81	2,115.81	0.00%

Table 4.73 – *Results for classical risk test case 3a*

shows the comparison of the OpenQuake result for expected loss over 75 years with the expected result.

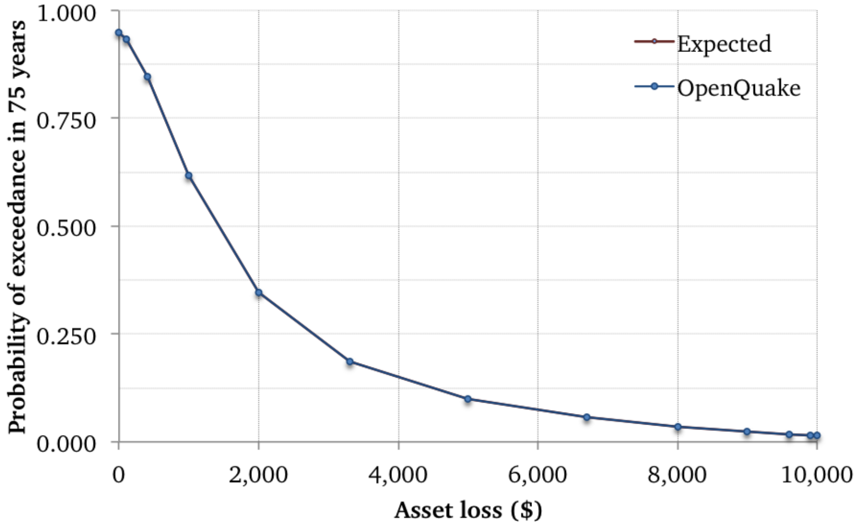

Figure 4.11 – *Loss curve comparison for classical risk test case 3a*

4.3.1.11 Case 4a

There are several ways by which the replacement value of an asset can be specified in the exposure model. The different options are listed below:

- Specify the aggregate value of each asset
- Specify the value per unit, and provide the number of units in each asset
- Specify the value per unit area, and provide the aggregate area of each asset
- Specify the value per unit area, specify the area per unit, and provide the number of units in each asset

This case tests the computation of the mean and standard deviation of the loss when the aggregate asset value is provided in the exposure model. The vulnerability function used is the same as in Case 1c and shown in Table 4.7. The aggregate asset value in this case is $20,000$. The average annual loss in this case should be exactly twice the value calculated in Case 1c. Table

Result	Expected	OpenQuake	Difference
Average loss	70.25	70.25	0.00%

Table 4.74 – *Results for classical risk test case 4a*

4.74 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.3.1.12 Case 4b

This case tests the computation of the loss exceedance curve and average annual loss when the value of the assets is specified per unit, and the number of units in each asset are provided in the exposure model. The vulnerability function used is the same as in Case 1c and shown in Table 4.7. The asset has two units, and the value per unit is $7,500$. The aggregate asset value in this case is $15,000$. Table 4.75 shows the comparison of the OpenQuake result for average

Result	Expected	OpenQuake	Difference
Average loss	52.69	52.69	0.00%

Table 4.75 – *Results for classical risk test case 4b*

annual loss with the expected result.

4.3.1.13 Case 4c

This case tests the computation of the loss exceedance curve and average annual loss when the value of the assets is specified per unit area, and the aggregate area of each asset is provided in the exposure model. The vulnerability function used is the same as in Case 1c and shown in Table 4.7. The asset has an aggregate area of 1,000 sq. units, and the value per unit area is 5. The aggregate asset value in this case is 5,000. The average annual loss in this case should be exactly half the value calculated in Case 1c. Table 4.76 shows the comparison of the OpenQuake result

Result	Expected	OpenQuake	Difference
Average loss	17.56	17.56	0.00%

Table 4.76 – *Results for classical risk test case 4c*

for average annual loss with the expected result.

4.3.1.14 Case 4d

This case tests the computation of the loss exceedance curve and average annual loss when the value of the assets is specified per unit area, the area is specified per unit, and the number of units in each asset are provided in the exposure model. The vulnerability function used is the same as in Case 1c and shown in Table 4.7. The asset has three units, the area per unit is 400 sq. units, and the value per unit area is 10. The aggregate asset value in this case is 12,000. Table 4.77

Result	Expected	OpenQuake	Difference
Average loss	42.15	42.15	0.00%

Table 4.77 – *Results for classical risk test case 4d*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.3.2 Multiple asset tests

The multiple asset test cases are designed to test the correct working of the classical risk calculator for a portfolio of assets of different taxonomies. The vulnerability functions for the different taxonomies are based on different intensity measure types.

The list of assets in the exposure model used for the multiple-asset test cases for the classical risk calculator is given in Table 4.29.

4.3.2.1 Case 5a

The purpose of this case is to test the basic elements of a classical risk calculation involving multiple assets, such as the computation of the loss exceedance curve and average annual loss for each asset in a portfolio of assets.

The list of assets and their taxonomies are shown in Table 4.29. The calculation of the loss curve and average annual loss for each of the seven assets follows the same procedure as described in Case 1c, and is not repeated here. Table 4.78 shows the comparison of the

Asset	Result	Expected	OpenQuake	Difference
a1	Average loss	35.07	35.07	0.00%
a2	Average loss	6.29	6.29	0.00%
a3	Average loss	1.36	1.36	0.00%
a4	Average loss	17.58	17.58	0.00%
a5	Average loss	7.87	7.87	0.00%
a6	Average loss	9.06	9.06	0.00%
a7	Average loss	11.38	11.38	0.00%

Table 4.78 – *Results for classical risk test case 5a*

OpenQuake results for average annual losses for the seven assets with the expected results.

4.3.3 Calculation with logic-trees

4.3.3.1 Case 7a

The OpenQuake scenario risk calculator allows the user to employ more than one ground motion prediction equation (GMPE) for computing the hazard curves used for the loss curve calculation. The loss exceedance curves, expected losses over the specified time period, and insured loss curves (if any), are calculated and output for each GMPE branch independently. No sampling is involved, and any branch weights assigned to the different GMPE branches are ignored. This case is designed to test the computation of the loss curve and average annual loss for an asset with two different GMPEs. The two ground motion prediction equations used are Boore and Atkinson (2008), and Chiou and Youngs (2008).

A single asset is used in this test case. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case. The loss curve calculation procedure follows the same steps as described earlier in Case 1c for each of the two individual hazard branches.

The expected loss curves for this case, computed using the procedure described in Case 1c, are compared with the loss curves obtained using the OpenQuake classical PSHA based risk calculator in Figure 4.12.

The area under the loss exceedance curves gives the average annual loss values for the two hazard branches.

Table 4.79 shows the comparison of the OpenQuake result for average annual losses for the two branches with the expected results.

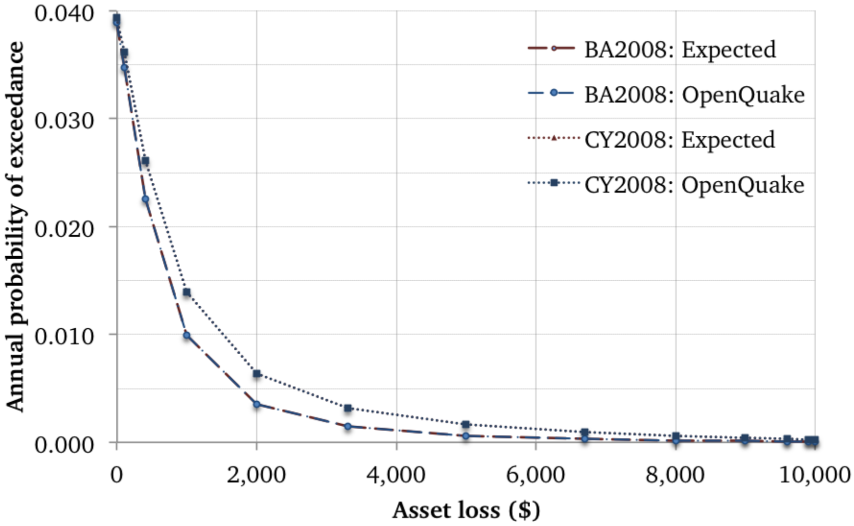

Figure 4.12 – *Loss curve comparison for classical risk test case 7a*

Branch	Asset	Result	Expected	OpenQuake	Difference
BA2008	a1	Average loss	35.13	35.13	0.00%
CY2008	a1	Average loss	49.55	49.55	0.00%

Table 4.79 – *Results for classical risk test case 7a*

4.4 Classical Damage Calculator

The tests for the classical PSHA-based damage calculator assume the correct computation of the hazard curves at the locations of the assets in the exposure model. Thus, the damage tests implicitly rely on the acceptance tests for the classical PSHA-based hazard calculator.

The source model used for the tests comprises a single vertical strike-slip fault with a Gutenberg-Richter b-value equal to 0.9 and a slip rate of 2 mm/yr. The MFD is a Gutenberg-Richter distribution truncated between magnitudes 5.0 and 6.5, while the Ground Motion Prediction Equation (GMPE) used is Boore and Atkinson (2008).

Details of the fault geometry are given below:

Fault type: Strike slip
Fault dip: 90°
Fault plane depths: 0–12 km
Fault coordinates:
South end: 38.0000°N, 122.0000°W
North end: 38.2248°N, 122.0000°W

The complete collection of input models and job configuration files used in these test cases can be accessed here: https://github.com/gem/oq-risklib/tree/master/openquake/qa_tests_data/classical_damage

4.4.1 Single asset tests

The single asset test cases are designed to test the basic elements of the classical-PSHA based damage calculator, such as:

- damage state occurrence probabilities for assets

The location and taxonomy of the single asset in the exposure model used for the single-asset test cases for the classical damage calculator are given in Table 4.1.

4.4.1.1 Case 1a

The purpose of this case is to test the integration of the derivative of the site hazard curve with the discrete fragility function for a single asset. Table 4.40 shows the set of ground motion intensity levels and corresponding probabilities of exceedance for the four damage states for the discrete fragility function used in this test case.

When the exposure model and discrete fragility model are provided to the OpenQuake classical PSHA-based hazard calculator, OpenQuake computes the hazard curves at the locations of the assets in the exposure model and at the specific intensity levels used in the fragility functions.

PGA	0.05g	0.20g	0.40g	0.60g	0.80g	1.00g	1.20g	1.40g	...	5.00g
P.O.E.	3.896×10^{-2}	2.222×10^{-2}	8.171×10^{-3}	3.070×10^{-3}	1.230×10^{-3}	5.195×10^{-4}	2.254×10^{-4}	9.918×10^{-5}	...	0.000

Table 4.80 – *Hazard curve for PGA at a single site*

The intensity levels for the hazard curve are extracted from the fragility function:
$[0.05, 0.20, 0.40, 0.60, 0.80, 1.00, 1.20, 1.40, \ldots, 5.00]$

The hazard curve gives the probabilities of exceedance for a set of intensity levels within a specified time period. The time period in this case, t_H, is one year. The hazard curve at the location of the single asset used in this test case is shown in Table 4.80.

The probabilities of exceedance are: $[3.986\times10^{-2}, 2.222\times10^{-2}, 8.171\times10^{-3}, 3.071\times10^{-3}, 1.230\times10^{-3}, 5.195\times10^{-4}, 2.254\times10^{-4}, 9.918\times10^{-5}, \ldots, 0]$. The probabilities of exceedance are first converted to annual rates (or frequencies) of exceedance by employing the Poissonion conversion:

$$\lambda(iml) = \frac{-\ln[1 - prob(IML > iml, t_H)]}{t_H} \tag{4.10}$$

The annual frequencies of exceedance are: $[3.974\times10^{2}, 2.247\times10^{2}, 8.205\times10^{3}, 3.075\times10^{3}, 1.231\times10^{3}, 5.197\times10^{4}, 2.254\times10^{4}, 9.918\times10^{5}, \ldots, 0]$.

The annual frequencies of occurrence are estimated by the differentiation of the annual frequencies of exceedance: $[1.727\times10^{2}, 1.426\times10^{2}, 5.130\times10^{3}, 1.845\times10^{3}, 7.109\times10^{4}, 2.942\times10^{4}, 1.262\times10^{4}, 5.565\times10^{5}, \ldots, 0]$.

Now, the annual frequencies of occurrence of the set of intensity measure levels, as computed above, are multiplied by the corresponding probabilities of exceedance of the damage states given these intensity measure levels. The probabilities of exceedance of the damage states are obtained directly from the discrete fragility function. For each damage state, the products thus obtained are summed across all the intensity measure levels, giving the annual frequency of exceedance for that damage state.

Assuming that exceedances of the damage states follow Poisson processes, the probabilities of exceedance for the set of damage states are calculated from the annual frequencies of exceedance $\lambda_{D \geq d_i}$ and the exposure time period t_R through:

$$prob(D \geq d_i, t_R) = 1 - \exp(-\lambda_{D \geq d_i} \times t_R) \tag{4.11}$$

Finally, the probabilities of occurrence for the set of damage states within the specified time period are computed from the probabilities of exceedance calculated above.

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	9.93×10^{-1}	9.93×10^{-1}	0%
	ds1	4.92×10^{-3}	4.92×10^{-3}	0%
	ds2	6.40×10^{-4}	6.40×10^{-4}	0%
	ds3	1.87×10^{-4}	1.87×10^{-4}	0%
	ds4	7.96×10^{-4}	7.96×10^{-4}	0%

Table 4.81 – *Results for classical damage test case 1a*

Table 4.81 shows the comparison of the OpenQuake damage state probabilities with the expected results.

4.4.1.2 Case 1b

Whereas the previous case was concerned with checking the correct implementation and usage of *discrete* fragility functions, the purpose of this case is to verify the correct calculation of damage distribution statistics for the classical damage calculator using *continuous* (lognormal CDF) fragility functions.

Table 4.42 shows the mean and standard deviation of the ground motion intensity level for the four damage states, which are the parameters for the lognormal fragility function used in this test case.

The only difference in the calculation procedure compared with Case 1a is the criterion used for selecting the set of intensity levels at which to compute the hazard curve. OpenQuake discretizes the continuous lognormal fragility functions into a set of intensity levels and corresponding probabilities of exceedance for each damage state. The minimum and maximum intensity levels used for this discretization are those specified in the fragility model file. The number of intervals into which this range of intensity levels is discretized is specified using the configuration parameter 'continuous_fragility_discretization'. In this case, the minimum and maximum intensity levels specifed in the continuous fragility model are $0.0g$ and $5.0g$ respectively. The parameter 'continuous_fragility_discretization' is set to 29, which means that OpenQuake will discretize the interval $[0.0, 5.0]g$ into 29 equal subintervals, yielding thirty intensity levels including the minimum and maximum limits.

The damage state exceedance probabilities are obtained by evaluating the complementary cumulative distribution function (CCDF) of the lognormal distribution at each of these thirty intensity levels, for the set of four damage states. Now, the calculation proceeds in the same manner as described in Case 1a above.

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	9.88×10^{-1}	9.88×10^{-1}	0%
	ds1	7.49×10^{-3}	7.49×10^{-3}	0%
	ds2	2.43×10^{-3}	2.43×10^{-3}	0%
	ds3	9.88×10^{-4}	9.88×10^{-4}	0%
	ds4	1.06×10^{-3}	1.06×10^{-3}	0%

Table 4.82 – *Results for classical damage test case 1b*

Table 4.82 shows the comparison of the OpenQuake damage state probabilities with the expected results.

4.4.1.3 Case 1c

Test Case 1c repeats the exercise from Case 1a, with the difference that the discrete fragility function specifies a minimum ground motion intensity, below which the probability of exceedance for all damage states is assumed to be zero.

Table 4.40 shows the discrete fragility function used in this test case. The "no damage limit" is specified to be $0.3g$. The procedure used for calculating the damage state probabilities remains the same as described in Case 1a, apart from this modification to the "no damage limit".

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	9.93×10^{-1}	9.93×10^{-1}	0%
	ds1	4.29×10^{-3}	4.29×10^{-3}	0%
	ds2	6.40×10^{-3}	6.40×10^{-3}	0%
	ds3	1.87×10^{-4}	1.87×10^{-4}	0%
	ds4	7.96×10^{-4}	7.96×10^{-4}	0%

Table 4.83 – *Results for classical damage test case 1c*

Table 4.83 shows the comparison of the OpenQuake damage state probabilities with the expected results.

4.4.1.4 Case 1d

Test Case 1d repeats the exercise from Case 1b, with the difference that the continuous fragility function specifies a minimum ground motion intensity, below which the probability of exceedance for all damage states is assumed to be zero.

Table 4.42 shows the continuous fragility function used in this test case. The "minimum intensity level" is specified to be $0.3g$. The procedure used for calculating the damage state probabilities remains the same as described in Case 1b, apart from this modification to the "minimum intensity level".

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	9.91×10^{-1}	9.91×10^{-1}	0%
	ds1	4.45×10^{-3}	4.45×10^{-3}	0%
	ds2	1.85×10^{-3}	1.85×10^{-3}	0%
	ds3	8.29×10^{-4}	8.29×10^{-4}	0%
	ds4	9.69×10^{-3}	9.69×10^{-3}	0%

Table 4.84 – *Results for classical damage test case 1d*

Table 4.84 shows the comparison of the OpenQuake damage state probabilities with the expected results.

4.4.1.5 Case 1e

This test case repeats the exercise from Case 1a using a discrete fragility model (Table 4.40) with a zero "no damage limit", and using four ‘steps_per_interval’. Each interval between the intensity levels specified in the fragility model is further divided into four equal subdivisions, thus ensuring a greater number of intensity levels at which the hazard curve will be computed. The procedure used for calculating the damage state probabilities remains the same as described in Case 1a, apart from this modification to the "steps_per_interval" interpolation parameter.

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	9.93×10^{-1}	9.93×10^{-1}	0%
	ds1	4.92×10^{-3}	4.92×10^{-3}	0%
	ds2	6.40×10^{-4}	6.40×10^{-4}	0%
	ds3	1.87×10^{-4}	1.87×10^{-4}	0%
	ds4	7.96×10^{-4}	7.96×10^{-4}	0%

Table 4.85 – *Results for classical damage test case 1e*

Table 4.85 shows the comparison of the OpenQuake damage state fractions with the expected results.

4.4.1.6 Case 1f

Case 1f is designed to test a simple multiplication of the damage state probabilities by the number of units comprising an asset. When an asset comprises more than one unit, the Classical Damage calculator returns the expected fraction of buildings in each damage state rather than the damage state probabilities. This case is thus designed identical to Case 1d, except that the single asset used in this case comprises three units, instead of one as in the previous case. The expected results should be three times those obtained in Case 1d.

Table 4.85 shows the comparison of the OpenQuake damage state fractions with the expected results.

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	2.97	2.97	0%
	ds1	1.33×10^{-2}	1.33×10^{-2}	0%
	ds2	5.55×10^{-3}	5.55×10^{-3}	0%
	ds3	2.49×10^{-3}	2.49×10^{-3}	0%
	ds4	2.91×10^{-3}	2.91×10^{-3}	0%

Table 4.86 – *Results for classical damage test case 1f*

4.4.1.7 Case 2a

The purpose of this case is to test the computation of the loss exceedance probabilities when the time period associated with the loss curve calculation is different from the time period associated with the hazard curve calculation. In this case, the hazard curve is calculated for a time period of 50 years, and the loss curve is calculated for a time period of 75 years. The procedure used for calculating the damage state probabilities remains the same as described in Case 1a, except that the values for the parameters t_H and t_R used in the equations to convert probabilities to frequencies and vice versa are $50yr$ and $75yr$ respectively in this case, whereas both parameters were set to $1yr$ in Case 1a.

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	6.11×10^{-1}	6.11×10^{-1}	0%
	ds1	2.74×10^{-1}	2.74×10^{-1}	0%
	ds2	4.36×10^{-2}	4.36×10^{-2}	0%
	ds3	1.31×10^{-2}	1.31×10^{-2}	0%
	ds4	5.80×10^{-2}	5.80×10^{-2}	0%

Table 4.87 – *Results for classical damage test case 2a*

Table 4.87 shows the comparison of the OpenQuake damage state probabilities with the expected results.

4.4.2 Multiple asset tests

The multiple asset test cases are designed to test the correct working of the classical damage calculator for a portfolio of assets of different taxonomies using different fragility functions.

4.4.2.1 Case 3a

The list of assets in the exposure model used in this case is given in Table 4.29. As shown in the table, the assets fall into three different taxonomies. Table 4.50 shows the parameters for the continuous lognormal fragility functions for the three taxonomies. Seven hazard curves are computed, one each for the seven sites. The procedure used for calculating the damage state probabilities for each of the seven individual assets remains the same as described in Case 1b, with the appropriate fragility function used for each asset depending on its assigned taxonomy. Table 4.88 shows the comparison of the OpenQuake results with the expected results for assets

Asset	Damage State	Expected	OpenQuake	Difference
a1	none	9.88×10^{-1}	9.88×10^{-1}	0%
	ds1	7.49×10^{-3}	7.49×10^{-3}	0%
	ds2	2.43×10^{-3}	2.43×10^{-3}	0%
	ds3	9.88×10^{-4}	9.88×10^{-4}	0%
	ds4	1.06×10^{-3}	1.06×10^{-3}	0%
a2	none	9.99×10^{-1}	9.99×10^{-1}	0%
	ds1	5.25×10^{-4}	5.25×10^{-4}	0%
	ds2	1.81×10^{-4}	1.81×10^{-4}	0%
	ds3	2.97×10^{-5}	2.97×10^{-5}	0%
	ds4	4.80×10^{-6}	4.80×10^{-6}	0%
a3	none	9.97×10^{-1}	9.97×10^{-1}	0%
	ds1	2.34×10^{-3}	2.34×10^{-3}	0%
	ds2	2.90×10^{-4}	2.90×10^{-4}	0%
	ds3	5.58×10^{-5}	5.58×10^{-5}	0%
	ds4	2.11×10^{-5}	2.11×10^{-5}	0%

Table 4.88 – *Results for classical damage test case 3a*

$a1$, $a2$, and $a3$.

4.4.3 Calculation with logic-trees

4.4.3.1 Case 4a

The OpenQuake scenario damage calculator allows the user to employ more than one ground motion prediction equation (GMPE) for computing the ground motion fields used for the loss calculation. The mean and standard deviation of the damage state probabilities (or fractions), are calculated and output for each GMPE branch independently. No sampling is involved, and any branch weights assigned to the different GMPE branches are ignored.

A single asset is used in this test case. The two ground motion prediction equations used are Boore and Atkinson (2008), and Chiou and Youngs (2008). Table 4.40 shows the parameters of the discrete fragility function used in this test case. Two hazard curves are computed, one for each of the GMPE branches specified in the logic tree. The procedure used for calculating the damage state probabilities for the asset remains the same as described in Case 1a. The damage state probabilities are computed individually for each of the two hazard branches.

Table 4.89 shows the comparison of the OpenQuake results for the damage distribution with the expected results for the two GMPE branches used in this case.

4.5 Event-Based Risk Calculator

The OpenQuake event based risk calculator builds on the outputs of the event based hazard calculator, employing Monte Carlo sampling techniques to compute event loss tables, asset loss exceedance curves, average annual asset losses, portfolio loss exceedance curves, and average

Branch	Asset	Damage State	Expected	OpenQuake	Difference
BA2008	a1	none	9.93×10^{-1}	9.93×10^{-1}	0%
		ds1	4.92×10^{-3}	4.92×10^{-3}	0%
		ds2	6.40×10^{-4}	6.40×10^{-4}	0%
		ds3	1.87×10^{-4}	1.87×10^{-4}	0%
		ds4	7.96×10^{-4}	7.96×10^{-4}	0%
CY2008	a1	none	9.90×10^{-1}	9.90×10^{-1}	0%
		ds1	7.03×10^{-3}	7.03×10^{-3}	0%
		ds2	1.24×10^{-3}	1.24×10^{-3}	0%
		ds3	3.76×10^{-4}	3.76×10^{-4}	0%
		ds4	1.61×10^{-3}	1.61×10^{-4}	0%

Table 4.89 – *Results for classical damage test case 4a*

annual portfolio loss. The tests for the event-based risk calculator assume the correct computation of the ground motion fields at the locations of the assets in the exposure model. Thus, the risk tests implicitly rely on the acceptance tests for the event-based hazard calculator.

4.5.1 Single asset tests

The single asset test cases are designed to test the basic elements of the event-based risk calculator, such as:

- asset event loss table computation
- asset loss exceedance curve computation

The location and taxonomy of the single asset in the exposure model used for the single-asset test cases for the event-based risk calculator are given in Table 4.1.

4.5.1.1 Case 1a

The source model used for this test case comprises a simple vertical strike-slip fault using a truncated Gutenberg-Richter magnitude-frequency distribution (MFD).

Details of the source model are listed below:

Fault type: Strike slip
Fault dip: 90°
Fault plane depths: 0–12 km
Fault coordinates:
South end: $38.0000°N$, $122.0000°W$
North end: $38.2248°N$, $122.0000°W$
Rupture aspect ratio: 2.0
Rake angle: 0°
Magnitude scaling relationship: Wells and Coppersmith (1994)
Magnitude-frequency distribution:

Truncated Gutenberg-Richter: $a = 3.1292$; $b = 0.9$; $M_{min} = 5.0$; $M_{max} = 6.5$

The time period for both the hazard and the risk calculations in this case are one year. This case uses a collection of 100,000 stochastic event sets, each spanning one year, to generate a set of ground motion fields representative of the seismicity of the specified region, collectively spanning a period of 100,000 years. A single stochastic event set (SES) may contain zero or more ruptures that are generated based on the frequency distribution of the sources. Each SES in this case represents one simulation of the possible events that might occur in one year.

Table 4.3 shows the mean loss ratios and corresponding coefficients of variation in the lognormal vulnerability function used in this case. There is no uncertainty in the vulnerability function used for this case. The coefficient of variation of the loss ratio is zero at all intensity measure levels. The purpose of this case is to test the correct interpolation of the mean loss ratios of the vulnerability function at intermediate intensity measure levels and verify the steps involved in the computation of the asset loss exceedance curve and average annual loss.

The hazard calculation produces 4,115 ruptures over the 100,000 cumulative time span, and 4,115 corresponding ground motion fields. The ground motion values at the location of the single asset are $[0.074, 0.154, 0.118, 0.203, \ldots, 0.288]g$ (4,115 ground motion values in total).

The calculation of the loss ratios given the ground motion values proceeds in exactly the same manner as described in the Scenario Risk calculator examples. The mean loss ratio and coefficient of variation of the loss ratio are obtained by linear interpolation from the provided vulnerability model for each of the above 4,115 ground motion values. Particularly in this case, since there is no variability in the loss ratio, calculation of the loss ratios for each ground motion field is straightforward. Since the coefficients of variation in the vulnerability function are all zero, the lognormal distribution devolves into the degenerate distribution. Thus, the sampled loss ratio at a particular ground motion intensity level is always equal to the mean loss ratio at that intensity level obtained through interpolation.

These numbers are multiplied by the asset value of $10,000$ to give 4,115 sampled loss values: $[148.90, 308.60, 236.14, 408.76, \ldots, 665.79]$ (4,115 loss values in total). This set of losses forms the event loss table (ELT) for the asset.

For the computation of the asset loss curve, the range of loss spanning the minimum and maximum loss observed in the ELT is divided into a number of equispaced intervals as specified by the parameter 'loss_curve_resolution'. In this case, a value of ten is used for the 'loss_curve_resolution' parameter. The maximum loss observed in the ELT is $9,713.29$, and the minimum loss observed is 0. The loss curve is thus calculated at the following eleven points: $[0.00, 1079.25, 2158.51, 3237.76, 4317.02, 5396.27, 6475.53, 7554.78, 8634.03, 9713.29]$.

At each of the eleven loss values on the loss curve, the annual frequency of exceedance of that loss value is obtained by counting the number of ruptures in the ELT which produce losses greater than that loss value, and dividing the count by the cumulative time span of 100,000 years. The loss exceedance counts for the eleven loss values shown above are: $[4030, 799, 293, 133, 87, 48, 25, 13, 6, 0]$. Dividing these exceedance counts by 100,000 gives the corresponding annual exceedance rates or frequencies: $[4.030 \times 10^{-2}, 7.990 \times 10^{-3}, 2.930 \times$

$10^{-3}, 1.330 \times 10^{-3}, 8.700 \times 10^{-4}, 4.800 \times 10^{-4}, 2.500 \times 10^{-4}, 1.300 \times 10^{-4}, 6.000 \times 10^{-5}, 0.000]$.

Finally, the probabilities of exceedance for the set of eleven loss values are calculated from the annual frequencies of exceedance $\lambda_{L \geq l}$ and the exposure time period t_R through:

$$prob(L \geq l, t_R) = 1 - \exp(-\lambda_{L \geq l} \times t_R) \tag{4.12}$$

The probabilities of exceedance for the eleven loss values are computed to be the following: $[3.950 \times 10^{-2}, 7.958 \times 10^{-3}, 2.926 \times 10^{-3}, 1.329 \times 10^{-3}, 8.696 \times 10^{-4}, 4.799 \times 10^{-4}, 2.500 \times 10^{-4}, 1.300 \times 10^{-4}, 6.000 \times 10^{-5}, 0.000]$.

The implementation of the event based risk calculator in Julia begins with the set of ground motion fields produced by the OpenQuake event based hazard calculator, and proceeds according to the steps listed above. The loss curve thus calculated above using Julia is compared with the loss curve obtained using the OpenQuake event-based risk calculator in Figure 4.13.

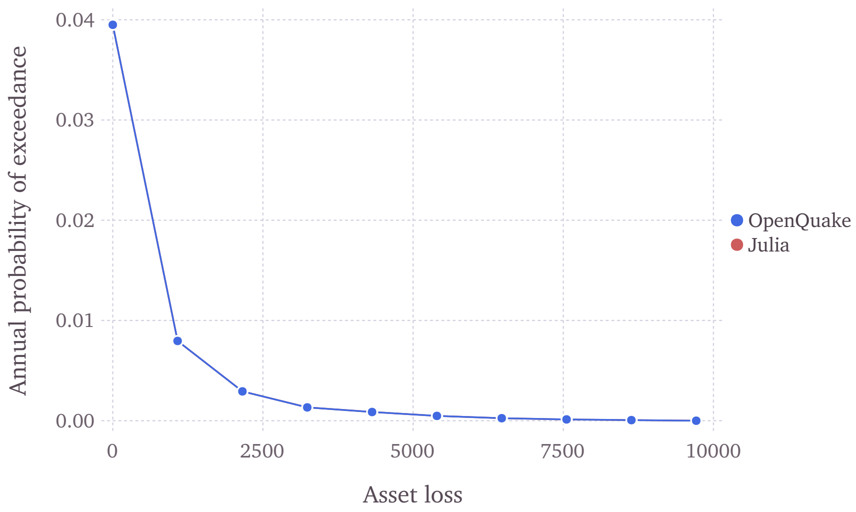

Figure 4.13 – *Loss curve comparison for event based risk test case 1a*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.90

Result	Expected	OpenQuake	Difference
Average loss	36.43	36.43	0.00%

Table 4.90 – *Results for event based risk test case 1a*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.2 Case 1b

The source model used for this test case comprises a simple vertical strike-slip fault using a single magnitude MFD.

Details of the source model are listed below:

Fault type: Strike slip
Fault dip: 90°
Fault plane depths: 0–1 km
Fault coordinates:
South end: 38.1124°N, 122.0450°W
North end: 38.1124°N, 121.9550°W
Rupture aspect ratio: 1.0
Rake angle: 0°
Magnitude scaling relationship: PEER MSR
Magnitude-frequency distribution:
Incremental MFD: $M_{min} = 4.0$; bin width = 1.0; occurance rate = 1.0

Apart from the different source model, this case is similar to Case 1a, and is only included here as the numbers from this case will be useful in later cases involving a nontrivial source model logic tree with more than branch. The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.14.

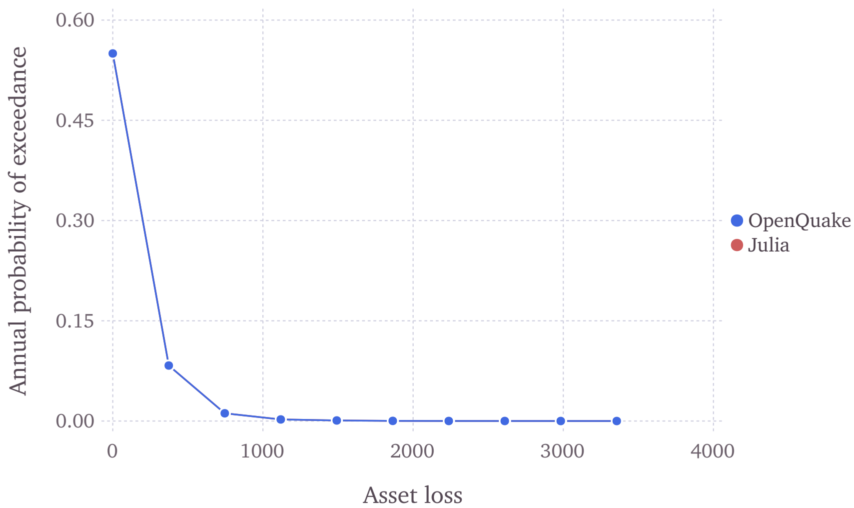

Figure 4.14 – *Loss curve comparison for event based risk test case 1b*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.91

Result	Expected	OpenQuake	Difference
Average loss	139.26	139.14	0.08%

Table 4.91 – *Results for event based risk test case 1b*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.3 Case 1c

The source model used for this test case comprises a characteristic fault.

Details of the source model are listed below:

Fault type: Strike slip
Fault dip: 30°
Fault plane depths: 5–15 km
Fault coordinates:
38.0000°N, 122.4000°W
38.2248°N, 122.0000°W
38.2248°N, 121.7000°W
Rake angle: 0°
Magnitude-frequency distribution:
Characteristic: $M = 7.0$; bin width = 0.1; occurance rate = 0.04

Apart from the different source model, this case is similar to Case 1a, and once again, is only included here as the numbers from this case will be useful in later cases involving a nontrivial source model logic tree with more than branch. The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.15.

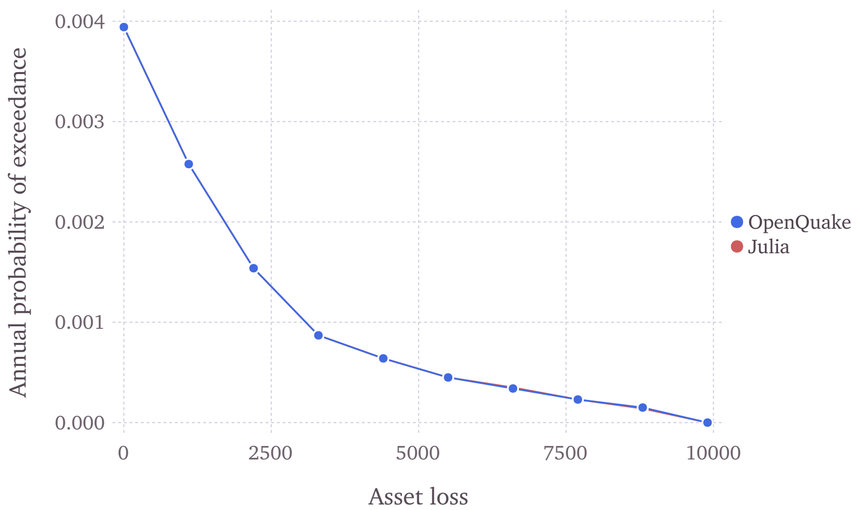

Figure 4.15 – *Loss curve comparison for event based risk test case 1c*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.92 shows the comparison of the OpenQuake result for average annual loss with the expected result.

Result	Expected	OpenQuake	Difference
Average loss	9.64	9.64	0.00%

Table 4.92 – *Results for event based risk test case 1c*

4.5.1.4 Case 1d

The purpose of this case is to test the loss ratio sampling when the lognormal vulnerability model has nonzero coefficients of variation of the loss ratios. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability model used in this test case.

Apart from the nonzero coefficients of variation in the vulnerability model, this case is similar to Case 1a. The only difference enters during the loss ratio sampling stage, where the lognormal distribution no longer devolves into the degenerate distribution.

As in Case 1a, the mean loss ratio and coefficient of variation of the loss ratio are obtained by linear interpolation from the provided vulnerability model for each of the 4,115 ground motion values produced by the hazard calculation. A loss ratio is now sampled from the lognormal distribution defined by the interpolated mean and standard deviation parameters for each ground motion value. The rest of the calculation steps involved in the computation of the event loss table, loss exceedance curve, and average annual loss remain the same as described in Case 1a.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.16.

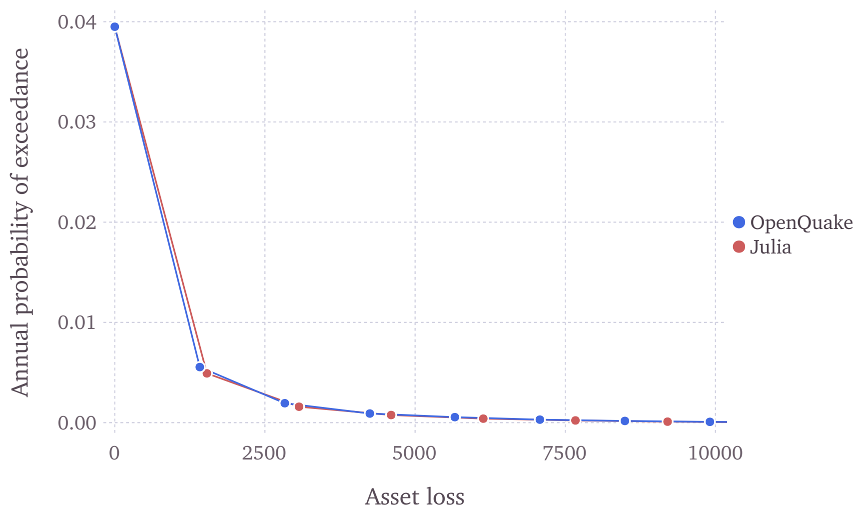

Figure 4.16 – *Loss curve comparison for event based risk test case 1d*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.93

Result	Julia	OpenQuake	Difference
Average loss	42.60	46.18	8.06%

Table 4.93 – *Results for event based risk test case 1d*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.5 Case 1e

This test case is identical to Case 1a described above, except for the use of the Beta distribution for the vulnerability functions instead of the lognormal distribution. Since the coefficients of variation in the vulnerability function are all zero, once again the Beta distribution devolves into the degenerate distribution as in Case 1a. The results for this test case should be exactly the same as in Case 1a.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.17.

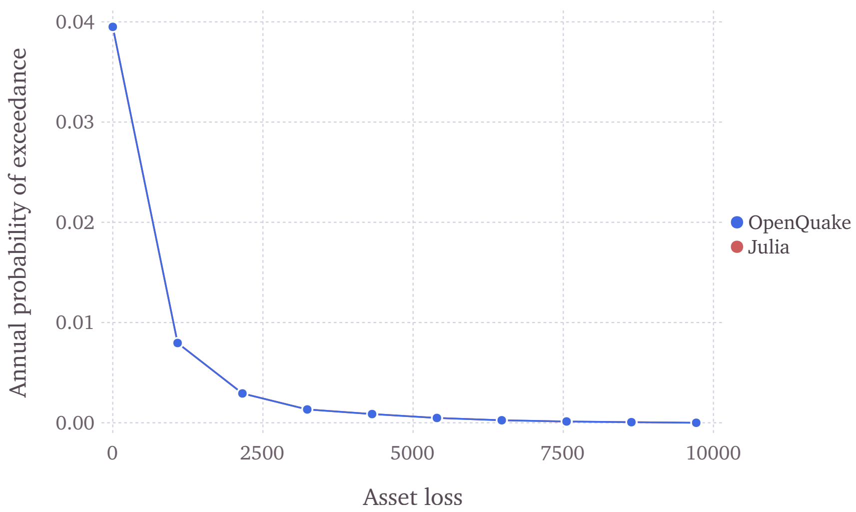

Figure 4.17 – *Loss curve comparison for event based risk test case 1e*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.94 shows the comparison of the OpenQuake result for average annual loss with the expected result.

Result	Expected	OpenQuake	Difference
Average loss	36.43	36.43	0.00%

Table 4.94 – *Results for event based risk test case 1e*

4.5.1.6 Case 1f

The purpose of this case is to test the loss ratio sampling when the Beta vulnerability model has nonzero coefficients of variation of the loss ratios. Table 4.7 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability model used in this test case.

Apart from the nonzero coefficients of variation in the vulnerability model, this case is similar to Case 1e. The only difference enters during the loss ratio sampling stage, where the Beta distribution no longer devolves into the degenerate distribution.

The area under the annual loss exceedance curve gives the average annual loss. Table 4.95 shows the comparison of the OpenQuake result for average annual loss with the expected result.

Result	Expected	OpenQuake	Difference
Average loss	36.80		%

Table 4.95 – *Results for event based risk test case 1f*

4.5.1.7 Case 1g

This test case repeats the exercise from Case 1d and Case 1f, using the discrete probability vulnerability functions instead of the parametric lognormal or Beta distribution based functions used in those two cases. The vulnerability model used in this test case is shown in Table 4.11. This vulnerability model specifies a set of loss ratios and the corresponding probabilities of occurrence for these loss ratios at different intensity measure levels.

In this case, for each simulated ground motion value, the probabilities of occurrence of the set of loss ratios used by the vulnerability function are obtained through interpolation as described earlier in Case 1c of the Scenario Risk Calculator. Using the set of loss ratios and the corresponding interpolated probabilities, one loss ratio is sampled for each ground motion value.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.18.

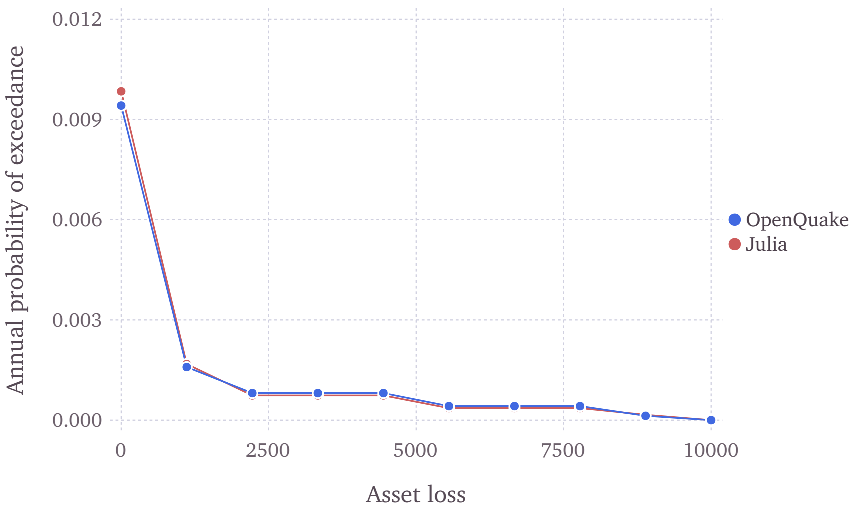

Figure 4.18 – *Loss curve comparison for event based risk test case 1g*

The area under the annual loss exceedance curve gives the average annual loss. Table 4.96

Result	Julia	OpenQuake	Difference
Average loss	11.17	11.23	0.53%

Table 4.96 – *Results for event based risk test case 1g*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.8 Case 2a

In addition to computing direct structural losses, OpenQuake also provides support for computing losses incurred for the following other loss types:

- Non-structural losses
- Contents losses
- Downtime, or business interruption losses
- Occupant fatalities

The purpose of this case is to test the calculation of loss curves for the non-structural components for an asset. The replacement value of the non-structural components for the asset used in this case is $15,000$. Table 4.17 shows the mean loss ratios and corresponding coefficients of variation in the non-structural components vulnerability function used in this test case. Apart from the change in the vulnerability function and value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.19.

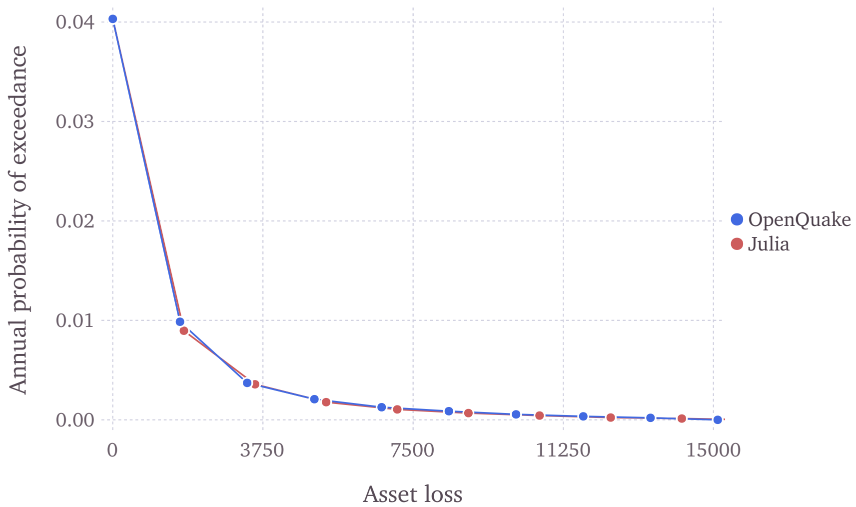

Figure 4.19 – *Loss curve comparison for event based risk test case 2a*

Result	Julia	OpenQuake	Difference
Average loss	65.70	66.53	1.26%

Table 4.97 – *Results for event based risk test case 2a*

Table 4.97 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.9 Case 2b

The purpose of this case is to test the calculation of loss curves and average loss for the contents of an asset. The replacement value of the contents for the asset used in this case is $5,000$.

Table 4.19 shows the mean loss ratios and corresponding coefficients of variation in the contents vulnerability function used in this test case. Apart from the change in the vulnerability function and value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.20.

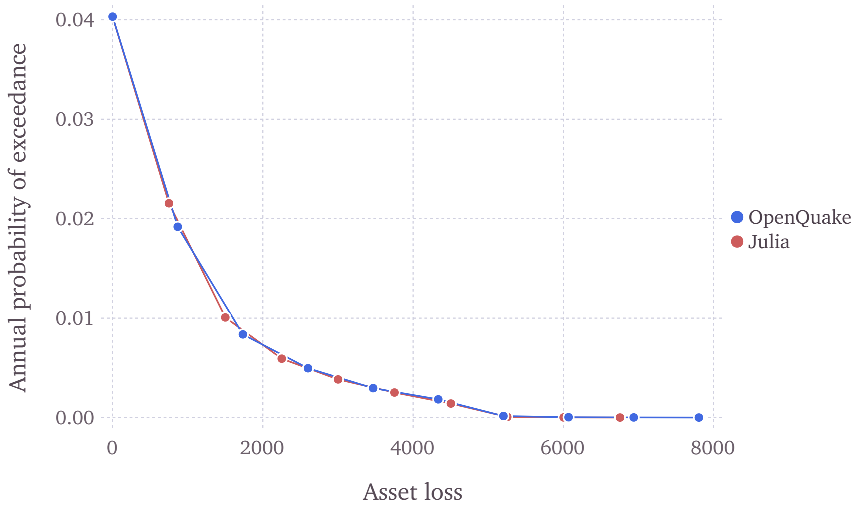

Figure 4.20 – *Loss curve comparison for event based risk test case 2b*

Result	Julia	OpenQuake	Difference
Average loss	49.17	49.44	0.55%

Table 4.98 – *Results for event based risk test case 2b*

Table 4.98 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.10 Case 2c

The purpose of this case is to test the calculation of exceedance curves for the downtime, or business-interruption losses for an asset. The loss due to downtime, or business-interruption for the asset used in this case is $2,000/month$. Downtime losses are usually specified per unit time the asset will be unavailable for occupancy or use. Table 4.21 shows the mean loss ratios and corresponding coefficients of variation for the downtime vulnerability function used in this test case. Apart from the change in the vulnerability function and value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.21.

Table 4.99 shows the comparison of the OpenQuake result for average annual loss with the expected result.

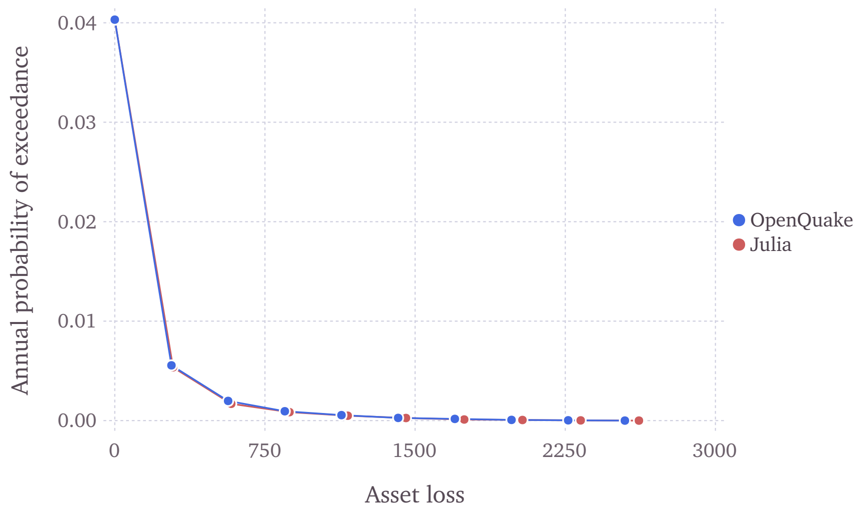

Figure 4.21 – *Loss curve comparison for event based risk test case 2c*

Result	Julia	OpenQuake	Difference
Average loss	8.43	8.28	1.81%

Table 4.99 – *Results for event based risk test case 2c*

4.5.1.11 Case 2d

The purpose of this case is to test the calculation of exceedance curves for occupant fatalities for an asset. The number of occupants for the asset used in this case are 2 (day), 4 (transit), and 6 (night). An average value of 4 occupants is used for the calculation of the exceedance curve and average annual fatalities. Table 4.21 shows the mean loss ratios and corresponding coefficients of variation for the occupants fatality vulnerability function used in this test case. Apart from the change in the vulnerability function and value, the calculation procedure remains the same as described in Case 1d.

Result	Julia	OpenQuake	Difference
Average fatalities	0.00023	0.00017	32.92%

Table 4.100 – *Results for event based risk test case 2d*

Table 4.100 shows the comparison of the OpenQuake result for average annual occupant fatalities with the expected result.

4.5.1.12 Case 3a

The purpose of this case is to test the computation of the loss exceedance probabilities when the time period associated with the loss curve calculation is different from the time period associated with the hazard curve calculation. In this case, each stochastic event set in the hazard calculation spans a time period of t_H =50 years, and the loss curve is calculated for a time period of t_R =75 years. The number of stochastic event sets (SES) used in this case is 20,000. The number of SES and t_H are employed in obtaining the annual loss exceedance rates or frequencies from the ELT, and t_R is employed as described in Equation 4.12 to convert back from rates of exceedance to probabilities of exceedance over the time period t_R.

Table 4.7 shows the mean loss ratios and corresponding coefficients of variation for the vulnerability function used in this test case.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.22.

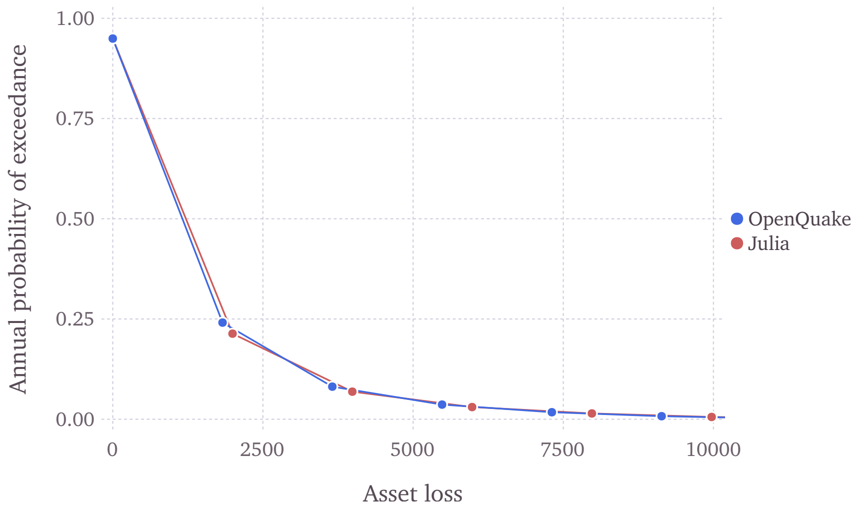

Figure 4.22 – *Loss curve comparison for event based risk test case 3a*

Result	Julia	OpenQuake	Difference
Average insured loss	1,614.06	1,573.71	2.53%

Table 4.101 – *Results for event based risk test case 3a*

Table 4.101 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.13 Case 4a

There are several ways by which the replacement value of an asset can be specified in the exposure model. The different options are listed below:

- Specify the aggregate value of each asset
- Specify the value per unit, and provide the number of units in each asset
- Specify the value per unit area, and provide the aggregate area of each asset
- Specify the value per unit area, specify the area per unit, and provide the number of units in each asset

This case tests the computation of the loss curve and average loss when the aggregate asset value is provided in the exposure model. The vulnerability function used is the same as in

Case 1d and shown in Table 4.7. The aggregate asset value in this case is 20,000. Apart from the change in the exposed value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.23.

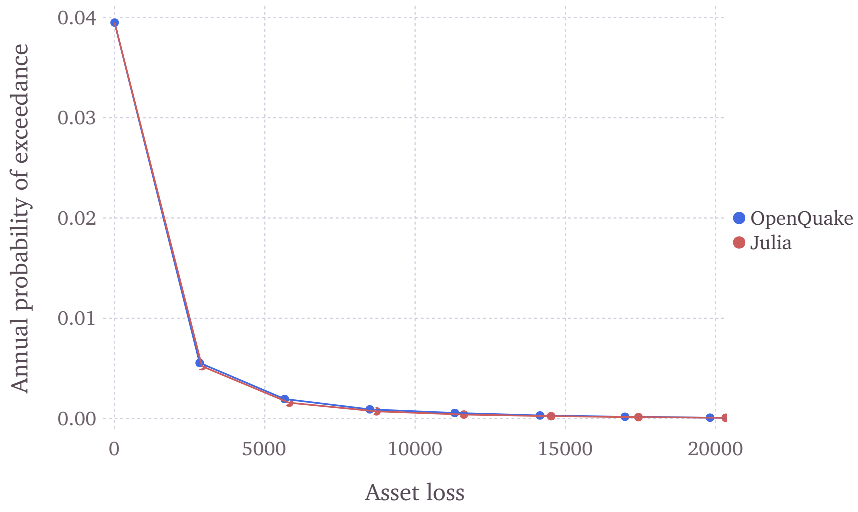

Figure 4.23 – *Loss curve comparison for event based risk test case 4a*

Result	Julia	OpenQuake	Difference
Average loss	81.65	82.86	1.47%

Table 4.102 – *Results for event based risk test case 4a*

Table 4.102 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.14 Case 4b

This case tests the computation of the loss curve and average loss when the value of the assets is specified per unit, and the number of units in each asset are provided in the exposure model. The vulnerability function used is the same as in Case 1d and shown in Table 4.7. The asset has two units, and the value per unit is 7,500. The aggregate asset value in this case is thus 15,000. Apart from the change in the exposed value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.24.

Result	Julia	OpenQuake	Difference
Average loss	70.73	69.13	2.29%

Table 4.103 – *Results for event based risk test case 4b*

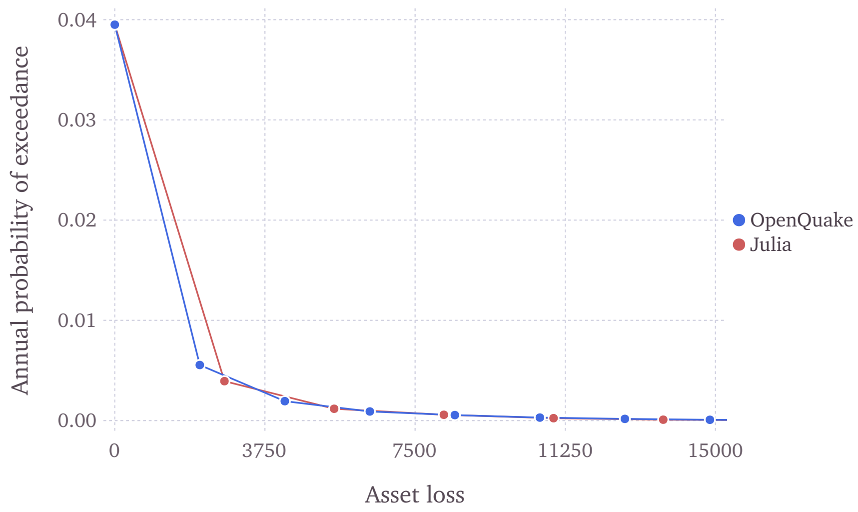

Figure 4.24 – *Loss curve comparison for event based risk test case 4b*

Table 4.103 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.15 Case 4c

This case tests the computation of the loss curve and average loss when the value of the assets is specified per unit area, and the aggregate area of each asset is provided in the exposure model. The vulnerability function used is the same as in Case 1f and shown in Table 4.7. The asset has an aggregate area of 1,000 sq. units, and the value per unit area is 5. The aggregate asset value in this case is thus 5,000. Apart from the change in the exposed value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.25.

Result	Julia	OpenQuake	Difference
Average loss	19.89	20.71	4.05%

Table 4.104 – *Results for event based risk test case 4c*

Table 4.104 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.16 Case 4d

This case tests the computation of the mean and standard deviation of the loss when the value of the assets is specified per unit area, the area is specified per unit, and the number of units in each asset are provided in the exposure model. The vulnerability function used is the same as in Case 1f and shown in Table 4.7. The asset has three units, the area per unit is 400 sq. units, and the value per unit area is 10. The aggregate asset value in this case is thus 12,000. Apart

Figure 4.25 – *Loss curve comparison for event based risk test case 4c*

from the change in the exposed value, the calculation procedure remains the same as described in Case 1d.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.26.

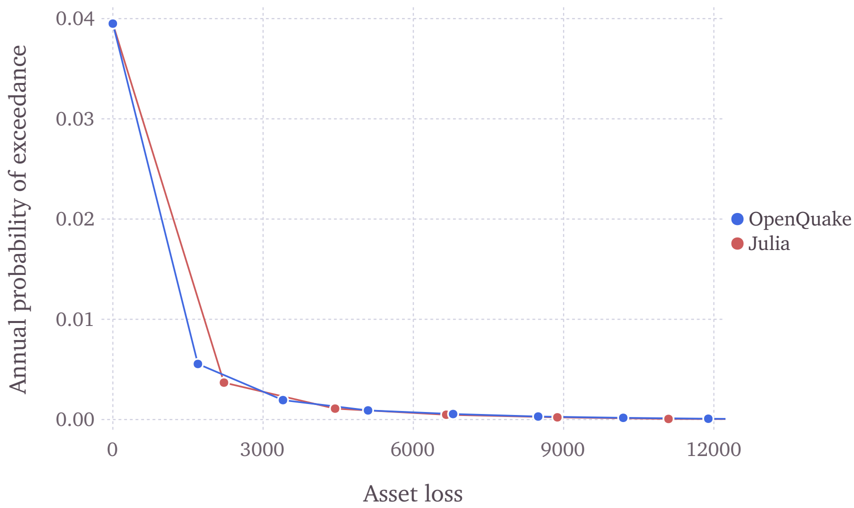

Figure 4.26 – *Loss curve comparison for event based risk test case 4d*

Table 4.105 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.1.17 Case 5a

Case 5a is a trivial test for loss curve calculation involving a source model consisting of multiple sources. This case assumes importance if the parallelization strategy of the hazard curve

Result	Julia	OpenQuake	Difference
Average loss	56.27	55.30	1.74%

Table 4.105 – *Results for event based risk test case 4d*

calculators is employed. Given that the source model consists of multiple sources, a task for each source can be defined, and therefore the task creation and aggregation of the results can be tested. The two sources comprising the single source model in this case are the ones described in Case 1a and Case 1b respectively. Given a reasonably large stochastic set of events, we should expect the rates of loss exceedance in this case to be the sum of the corresponding rates from Case 1a and Case 1b.

There is no uncertainty in the vulnerability function used for this case. Table 4.3 shows the mean loss ratios and corresponding coefficients of variation in the lognormal vulnerability function used in this case.

The loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.27.

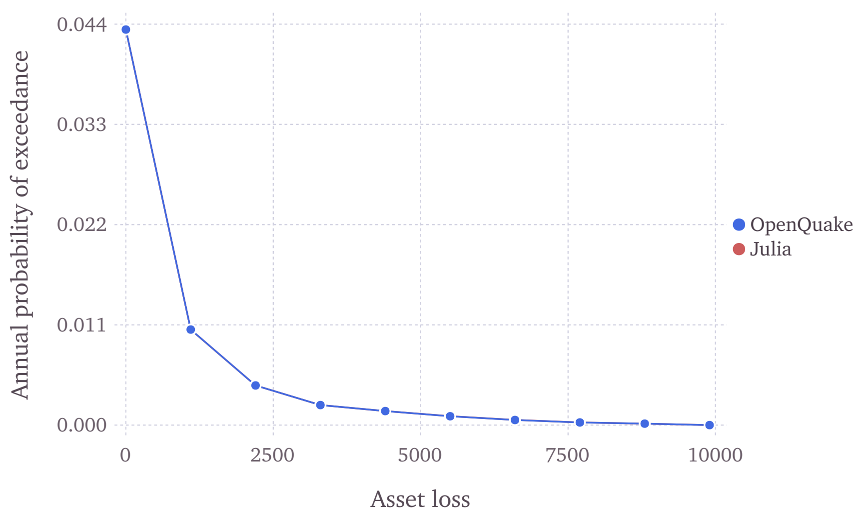

Figure 4.27 – *Loss curve comparison for event based risk test case 5a*

Result	Julia	OpenQuake	Difference
Average loss	46.49	46.48	0.02%

Table 4.106 – *Results for event based risk test case 5a*

Table 4.106 shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.2 Multiple asset tests

The multiple asset test cases are designed to test the loss aggregation functions of the event-based risk calculator, such as:

- portfolio loss computation for a given ground motion field
- calculation of portfolio loss exceedance curves

The major differences from the single asset calculations that need to be considered in the multiple asset tests include the possibility of having spatially-correlated ground motion fields and correlated vulnerability models for different assets of the same taxonomy.

4.5.2.1 Case 6a

The purpose of this case is to test the basic elements of an event based risk calculation involving multiple assets, such as computation of the individual asset loss curves, the portfolio loss exceedance curve, average asset losses, and the average portfolio loss.

The list of assets and their taxonomies are shown in Table 4.34, and Table 4.3 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

As in previous cases, ground motion fields are generated for each of the ruptures generated in the 100,000 stochastic event sets. These ground motion fields take into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied. These ground motion fields are also used for the corresponding calculation in Julia.

Since there is no variability in the loss ratio, calculation of the loss ratios is straightforward in this case. The loss tables for each of the seven assets in the portfolio is compiled in the same manner as described in the single asset Case 1a. Since the coefficients of variation in the vulnerability function are all zero, the lognormal distribution devolves into the degenerate distribution. Thus, the sampled loss ratio at a particular ground motion intensity level is always equal to the mean loss ratio at that intensity level obtained through interpolation. Since there is no random sampling of the loss ratios in this case, we should expect to find the results from OpenQuake and Julia to be exactly identical.

The portfolio loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.28.

Table 4.107 shows the comparison of the OpenQuake result for average asset losses and average portfolio loss with the expected result.

4.5.2.2 Case 6b

This case involving multiple assets is designed to test of the computation of the individual asset loss curves, the portfolio loss exceedance curve, average asset losses, and the average portfolio loss, when the vulnerability models of different assets of the same taxonomy are treated as uncorrelated. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter 'asset_correlation' to zero.

The list of assets and their taxonomies are shown in Table 4.34. Table 4.31 shows the mean

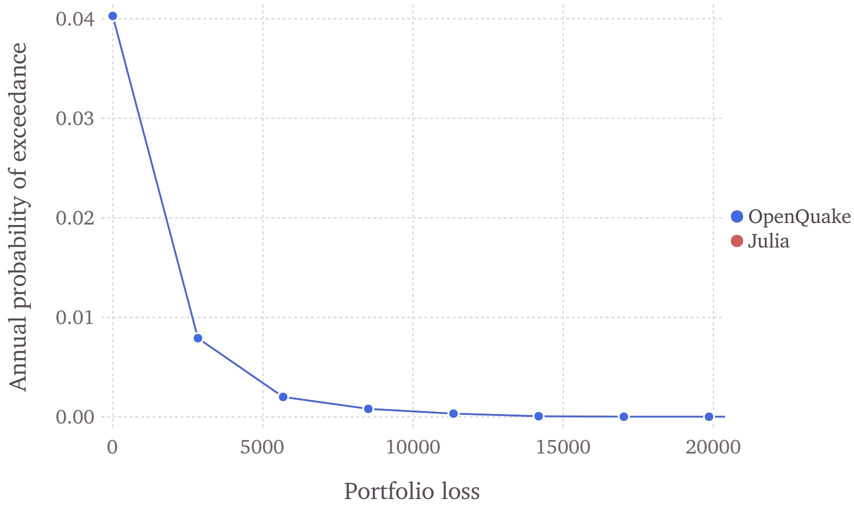

Figure 4.28 – *Portfolio loss curve comparison for event based risk test case 6a*

Asset	Result	Julia	OpenQuake	Difference
a1	Average loss	35.61	35.61	0.00%
a2	Average loss	8.12	8.12	0.00%
a3	Average loss	0.71	0.71	0.00%
a4	Average loss	22.25	22.25	0.00%
a5	Average loss	4.83	4.83	0.00%
a6	Average loss	17.16	17.16	0.00%
a7	Average loss	7.64	7.64	0.00%
Portfolio	Average loss	88.84	88.84	0.00%

Table 4.107 – *Results for event based risk test case 6a*

loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Ground motion fields are generated for each of the ruptures generated in the 100,000 stochastic event sets. These ground motion fields take into consideration both the inter-event and intra-event variability in the ground motion. The ground motion prediction equation used is Boore and Atkinson (2008), and the Jayaram and Baker (2009) model for spatial correlation of ground motion values is applied. These ground motion fields are also used for the corresponding calculation in Julia.

Since the sampled loss ratios conditional on a given ground motion field for different assets of the same taxonomy are assumed to be uncorrelated in this case, a loss ratio is sampled independently for each asset from the univariate lognormal distribution for that asset for each ground motion field.

The portfolio loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.29. Only the aggregated results for the portfolio are shown here for brevity.

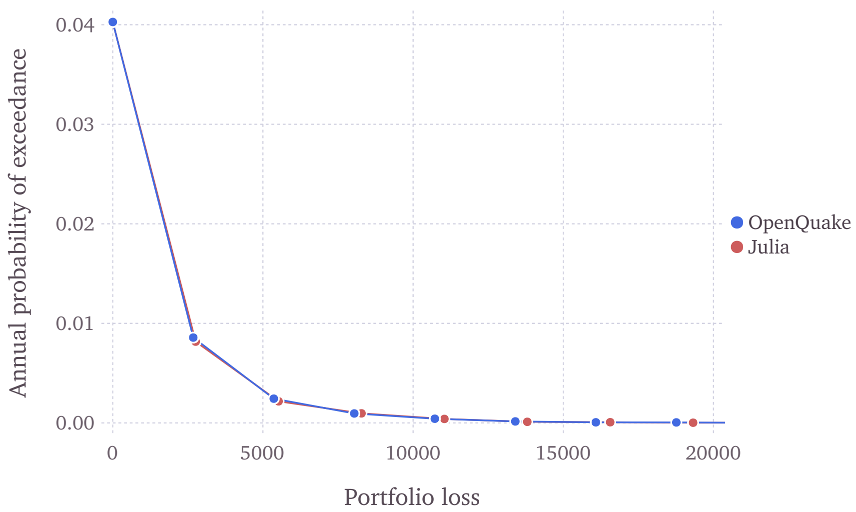

Figure 4.29 – *Portfolio loss curve comparison for event based risk test case 6b*

Asset	Result	Julia	OpenQuake	Difference
Portfolio	Average loss	88.48	87.89	0.66%

Table 4.108 – *Results for event based risk test case 6b*

Table 4.108 shows the comparison of the OpenQuake result for average portfolio loss with the expected result.

4.5.2.3 Case 6c

This multiple asset case is designed to test the computation of the individual asset loss curves, the portfolio loss exceedance curve, average asset losses, and the average portfolio loss, when the vulnerability models of different assets of the same taxonomy are treated as fully correlated. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter 'asset_correlation' to one.

The list of assets and their taxonomies are shown in Table 4.34. Table 4.31 shows the mean loss ratios and corresponding coefficients of variation in the vulnerability function used in this test case.

Ground motion fields are generated for each of the ruptures generated in the 100,000 stochastic event sets as described in Case 6a and Case 6b. These ground motion fields are also used for the corresponding calculation in Julia.

Since the sampled loss ratios conditional on a given ground motion field for different assets of the same taxonomy are assumed to be fully correlated in this case, a single *epsilon*, ε, is sampled from the standard normal distribution for each taxonomy. The parameters m and s from the vulnerability model are converted to the parameters μ and σ of the corresponding normal distribution, and the sampled loss ratio is obtained simply as $\exp(\mu + \varepsilon * \sigma)$.

The portfolio loss curve calculated using the implementation of the calculator in Julia is compared with that produced by OpenQuake in Figure 4.30. Only the aggregated results for the portfolio are shown here for brevity.

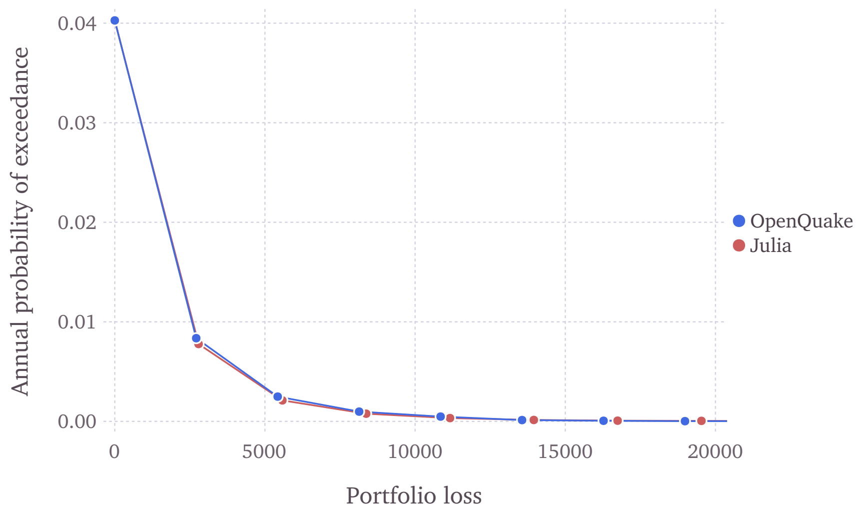

Figure 4.30 – *Portfolio loss curve comparison for event based risk test case 6c*

Asset	Result	Julia	OpenQuake	Difference
Portfolio	Average loss	87.61	88.61	1.13%

Table 4.109 – *Results for event based risk test case 6c*

Table 4.109 shows the comparison of the OpenQuake result for average portfolio loss with the expected result.

4.5.3 Insurance tests

The insurance test cases are designed to test the insurance related elements of the event-based risk calculator, such as:

- asset insured loss table computation
- asset insured loss exceedance curve computation
- portfolio insured loss table computation
- portfolio insured loss exceedance curve computation

4.5.3.1 Case 7a

In addition to calculating individual asset and portfolio loss tables and exceedance curves, OpenQuake also calculates the insured losses and exceedance curves for individual assets and for the portfolio, if requested. Each loss type can be assigned a deductible and an insurance limit, specified in either relative or absolute terms with respect to the replacement value of the asset.

This purpose of this case is to test the computation of individual asset insured loss exceedance curves and the average insured loss for each asset.

Given an asset loss $loss$, and a deductible component of insurance $deductible$, and an insurance limit $limit$, the insured loss is zero if the asset loss is below the deductible. For losses above the deductible amount, insurance pays the difference up to the limit. The insured loss is thus the smaller of the difference and the insurance limit. The equation used for computing the insured loss is presented below:

$$insured\_loss = min(max(loss - deductible, 0.0), limit - deductible) \tag{4.13}$$

The insured asset losses are collected for each of the ground motion fields generated for the simulated stochastic event sets, and finally the exceedance curves for the insured losses are calculated.

The input models for this test case are based on those used earlier in the single asset test Case 1d. The deductible component of insurance is $0.1\times$ the cost of replacement of the asset. The insurance limit is capped at $0.8\times$ the cost of replacement of the asset. Table 4.110 shows

Result	Julia	OpenQuake	Difference
Average asset loss	44.50		%
Average asset insured loss	11.46		%

Table 4.110 – *Results for event based risk test case 7a*

the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.3.2 Case 7b

This case is designed to exercise the portfolio insured loss exceedance curve computation for an exposure model containing multiple assets. In this case, the vulnerability models of different assets of the same taxonomy are treated as uncorrelated. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter ‘asset_correlation’ to zero.

The sampling of the loss ratios proceeds as described earlier in Case 6b, and the insured losses for the individual assets are obtained from the asset event loss tables as described in Case 7a. Apart from the individual asset insured loss exceedance curves, in this case, the portfolio insured loss exceedance curve is also computed. Finally, the average portfolio insured loss is computed as the area under the portfolio insured loss exceedance curve. Table 4.111

Result	Julia	OpenQuake	Difference
Average portfolio loss	90.75		%
Average portfolio insured loss	19.43		%

Table 4.111 – *Results for event based risk test case 7b*

shows the comparison of the OpenQuake result for average annual loss with the expected result.

4.5.3.3 Case 7c

This case is designed to exercise the portfolio insured loss exceedance curve computation for an exposure model containing multiple assets. In this case, the vulnerability models of different assets of the same taxonomy are treated as fully correlated. In OpenQuake, this can be specified in the job configuration file, by setting the value of the parameter 'asset_correlation' to one.

The sampling of the loss ratios proceeds as described earlier in Case 6c, and the insured losses for the individual assets are obtained from the asset event loss tables as described in Case 7a. Apart from the individual asset insured loss exceedance curves, in this case, the portfolio insured loss exceedance curve is also computed. Finally, the average portfolio insured loss is computed as the area under the portfolio insured loss exceedance curve. Table 4.112 shows the comparison

Result	Julia	OpenQuake	Difference
Average portfolio loss	88.10		%
Average portfolio insured loss	18.87		%

Table 4.112 – *Results for event based risk test case 7c*

of the OpenQuake result for average annual loss with the expected result.

Bibliography

Berkes, P. (2012). *Writing robust scientific code with testing (and Python)*. Euroscipy. URL: http://archive.euroscipy.org/talk/6634 (cited on page 14).

Boore, D. M. and G. M. Atkinson (2008). "Ground-Motion Prediction Equations for the Average Horizontal Component of PGA, PGV, and 5%-Damped PSA at Spectral Periods between 0.01 s and 10.0 s". In: *Earthquake Spectra* 24.1, pages 99–138 (cited on pages 35, 36, 41, 43, 45, 47, 55, 57, 61, 63, 64, 76, 77, 83, 100, 102).

Chiou, B. S.-J. and R. R. Youngs (2008). "An NGA Model for the Average Horizontal Component of Peak Ground Motion and Response Spectra". In: *Earthquake Spectra* 24, pages 173–215 (cited on pages 47, 63, 76, 83).

Cornell, C. A. (1968). "Engineering Seismic Risk Analysis". In: *Bulletin of the Seismological Society of America* 58, pages 1583–1606 (cited on page 27).

Jayaram, N. and J. W. Baker (2009). "Correlation model for spatially distributed ground-motion intensities". In: *Earthquake Engineering and Structural Dynamics* 38.15, pages 1687–1708. DOI: 10.1002/eqe.922 (cited on pages 41, 43, 45, 57, 61, 100, 102).

McGuire, K. K. (1976). *FORTRAN computer program for seismic risk analysis*. Open-File report 76-67. 102 pages. United States Department of the Interior, Geological Survey (cited on page 27).

Myers, G. J., C. Sandler, and T. Badgett (2012). *The art of software testing*. Wiley and Sons, Inc. (cited on pages 13, 19).

Thomas, P., I. Wong, and N. A. Abrahamson (2010). *Verification of Probabilistic Seismic Hazard Analysis Computer Programs*. PEER Report 2010/106. College of Engineering, University of California, Berkeley: Pacific Earthquake Engineering Centre (cited on page 64).