HazImp Documentation

Release 0.2

Duncan Gray

April 20, 2015

CONTENTS

HazImp is a tool for determining the impact due to natural hazards. It can be used to calculate damage to sites, given exposure and hazard information. It is command line based and can be executed in parallel.

CHAPTER

ONE

USER GUIDE

1.1 Introduction

HazImp is used to simulate the loss of value to structures from natural hazards using vulnerability curves. Generally the input information is hazard, such as a wind speed raster and exposure. The exposure information is currently supplied as a csv file, with structure locations given in latitude and longitude. This is combined with vulnerability curve information, described in an xml file. Figure 1.1 is an example of a vulnerability curve, showing a hazard value of the x-axis and the loss associated with that hazard on the y-axis;

1.2 Quick how-to

Follow the install notes in the README.md file.

A configuration file can be used to define a HazImp simulation. The configuration file is described using yaml, a data serialisation format. HazImp can also be used by another Python application, by passing the configuration infomation in as a dictionary.

For example, to run a wind example do;:

```
cd examples/wind
python ../../core_hazimp/hazimp.py  -c wind_v3.yaml
```

The -c specifies the configuration file.

HazImp can also be ran in parallel, using mpirun. For example;:

```
mpirun -np 4 python ../../core_hazimp/hazimp.py  -c wind_v3.yaml
```

There are a suite of HazImp tests to test the install and code during software developemnt. To run these, in the root HazImp directory do;:

```
./all_tests
```

1.3 Templates

The simplest way to use HazImp is with a template. There is currently a wind template. Templates take into account internal vulnerability curves and the data flow needed to produce loss information, simplifying the configuration file.

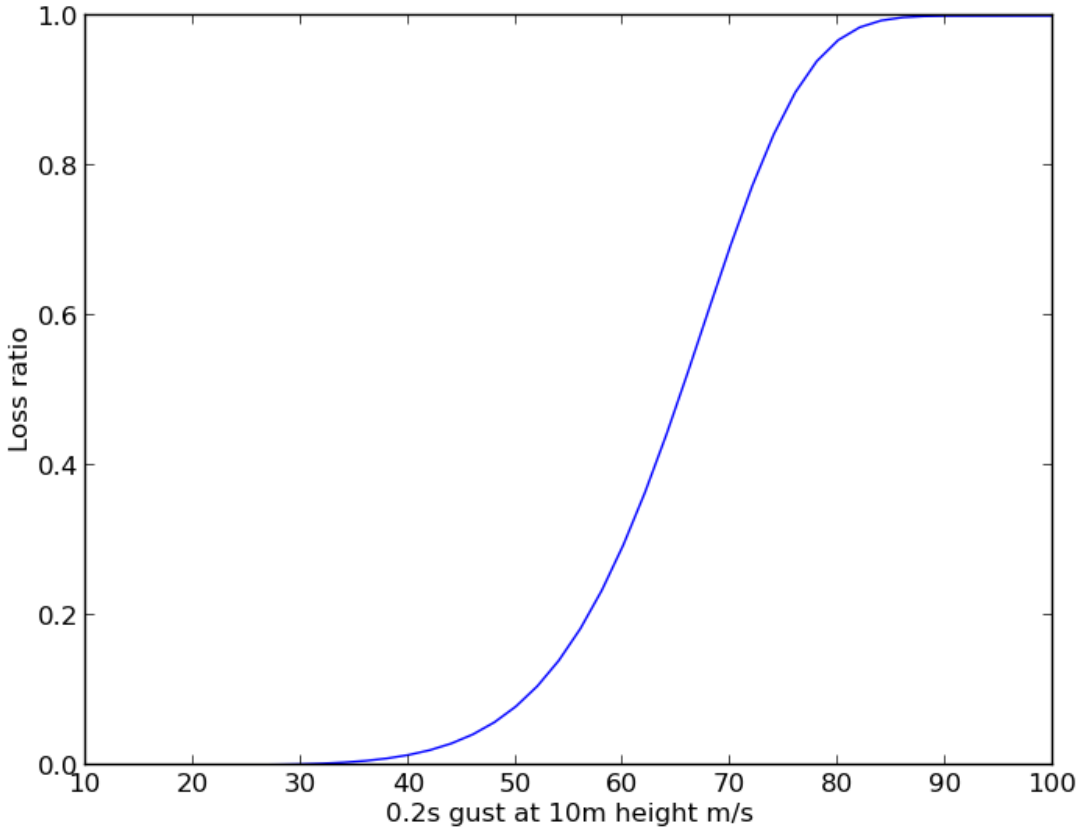

Figure 1.1: *An example vulnerability curve.*

1.4 Wind Template

Given gust information from TCRM and point exposure data the loss associated with each site is calculated using the wind template. The wind vulnerability functions That are used are built-in to HazImp. They are defined in the GA internal report

Here is the example wind configuration file (from examples/wind), which uses the wind template.:

```
#  python hazimp.py -c wind_v3.yaml
 - template: windv3
 - load_exposure:
    file_name: WA_Wind_Exposure_2013_Test_only.csv
    exposure_latitude: LATITUDE
    exposure_longitude: LONGITUDE
 - load_wind_ascii: [gust01.txt, gust02.tx]
 - calc_struct_loss:
    replacement_value_label: REPLACEMENT_VALUE
 - save: wind_impact.csv
```

The first line is a comment, so this is ignored. The rest of the file can be understood by the following key value pairs;

template The type of template to use. This example describes the *windv3* template.

load_exposure This loads the exposure data. It has 3 sub-sections;

> ***file_name*** The name of the csv exposure file to load. The first row of the csv file is the title row.
>
> ***exposure_latitude*** The title of the csv column with latitude values.
>
> ***exposure_longitude*** The title of the csv column with longitude values.

load_wind_ascii A list of ascii grid wind hazard files to load or a single file. The file format is grid ascii. The values in the file must be `0.2s gust at 10m height m/s`, since that is the axis of the HazImp wind vulnerability curves.

calc_struct_loss

> This will multiply the replacement value and the `structural_loss_ratio` to get the `structural_loss`.
>
> ***replacement_value_label*** The title of the exposure data column that has the replacement values.

save The file where the results will be saved. All the results to calculate the damage due to the wind hazard are saved to file. The above example saves to a csv file, since the file name ends in *.csv*. This has the disadvantage of averaging data from multiple wind hazards. The information can also be saved as numpy arrays. This can be done by using the *.npz* extension. This data can be accessed using Python scripts and is not averaged.

> There are some pre-requisites for the exposure data. It must have a column called `WIND_VULNERABILITY_FUNCTION_ID` which describe the vulnerability functions to be used. The vulnerability set used is hard coded. This is used to calculate the `structural_loss_ratio` given the `0.2s gust at 10m height m/s`.

1.5 Flood Template - Structural Damage

The structural damage flood template is very similar to the the wind template. This is an example structural damage flood template;:

```
#  python ../../core_hazimp/hazimp.py -c list_flood_v2.yaml
# Don't have a scenario test automatically run this.
# Since the file location is not absolute,
- template: flood_fabric_v2
- floor_height_(m): .3
- load_exposure:
    file_name:  small_exposure.csv
    exposure_latitude: LATITUDE
    exposure_longitude: LONGITUDE
- load_flood_ascii:  depth_small_synthetic.txt
- calc_struct_loss:
    replacement_value_label: REPLACEMENT_VALUE
- save: flood_impact.csv
```

The first 4 lines are comments, so they are ignored. The new key value pairs are;

floor_height_(m) This is used to calculate the water depth above ground floor; water depth(m) - floor height(m) = water depth above ground floor(m)

load_flood_ascii A list of ascii grid hazard files to load or a single file. The file format is grid ascii. The values in the file must be `water depth(m)`, since that is the axis of the vulnerability curves.

1.6 Without Templates