UNIVERSITÀ
DEGLI STUDI DI TRIESTE

dipartimento di matematica e geoscienze

The DMG Manuals

Neodeterministic hazard computation for tsunami

2D models

University of Trieste
Department of Mathematic and Geosciences
Group of Seismology
via Weiss, 4
34128 Trieste, Italy

Document name: DSTXTsunamiHazardScenario2D
Document Version: 1.1
Date Issued: September 26, 2017 3:48 PM
Author: Davide Bisignano

Table of Contents

Tsunami hazard computation in 2D models

Introduction

This is a step by step manual to tell you how to use the suite of programs called “haztsu2d” designed of the computation of neodeterministic hazard scenario for tsunami in laterally heterogeneous models. n particular the path between is modeled choosing two different structures, one at the source and one at the receiver. In this way the formal develop of the routine is identical to that for 1D case and we can calculate a lot of signals (necessary to establish an hazard scenario) with low computational cost. You can use different source-receiver configurations: default cases have fixed sources (from file .sut) and fixed receivers (file .obs) or a grid of receiver. In these two cases, at the state of the art, the source is considered pointlike but there is also the possibility of a different run in which extended models (treated by program PULSYN by Gusev) are used. At the moment the extended source can ben used only for fixed receivers and not for a receivers grid. There is also the possibility to obtain animation of the tsunami wave propagation. For further explanations or suggestions to improve this manual contact Davide Bisignano.

The suite consists of 10 programs :

makehaztsu2d.f: prepares the script for automatic run of the programs sequence.
tsuhazcpt.f: prepares the color palettes for plotting
polabel.f: create the label files for the polygons file.
tsupatgen.f: defines sources-receivers paths
tsuinput.f: prepares input file for seismograms computation.
syt2d.f: computes seismograms
efft.f: processes seismograms (ask Franco for details)
couanim.f: extract relevant parameters from signals and in case of animation prepares the necessary files.
tsufinmax.f: selects for each files the value to be reported on the maps, in tsunami case this value is the maximum vertical displacement of the wave
tsumaxsig.f: selects for each site the seismogram with the maximum peak.

How to proceed. Default run

Preparing and execute the script

To start the procedure with the default parameters you have to run the program

```
makehaztsu2d.out
```

and then execute the script haztsu by typing:

```
sh haztsu
```

List of required input files

Before running the programs you must have in your folder the following files listed below. Each run must be uniquely identified by a sequence of characters (root) in the filename. All the examples in this manual refer to a test run with root "xxx". Since rather long filenames are generated by the programs, it is strongly suggested that the root does not exceed 30 characters length.

makehaz.par: parameter files that creates the script that activate the flow of programs
cells.par: the parameter file for the first program of the sequence
xxx.fps: the fault plane solution available
yyy.dat: the earthquake catalog with historical seismicity
yyy.poc: the polygons defining the area where yyy.dat has to be used
xxx.por: the polygons associated with average regional structural models
xxx.pos: the polygons that define the seismogenic zones
xxx.sut: file with source information
xxx.obs: file with receiver information
xxx0001.spt: file with spectrum associated with polygon 0001 defined in xxx.por
.
.
.
xxx000n.spt: the file with the spectrum associated with polygon 000n in xxx.por
tsugmt.sh: script for plotting
gusev01.xy: spectral curve for scaling seismograms with magnitude
.
.
.
gusev10.xy: spectral curve for scaling seismograms with magnitude
itacode.dat: file with code response spectra for computing DGA

Makehaz.par: parameter file for the program that creates the hazard sequence

```
Parameters for program makehaz
---------------------------------------------------------------
RUN DEFINITION
---------------------------------------------------------------
xxx                                  Name of the run (max 15 char.)
   6 20           Min and max longitude              (degrees)
  36 48           Min and max latitude               (degrees)
1                 Use seismogenic zones (0=no, 1=yes)
0                 Use nodes (0=no, 1=yes, 2= yes separately)
0                 File with nodes coordinates (max 12 char.)(for nodes only)
0                 Use alerted areas (0=no, 1=CN, 2=M8S)
0                 Execution (0=full ,1=until sources,2=until paths,3=starting from
obs+sut,4=sut+grid)
1                 Clean level (0=no,1=yes)
---------------------------------------------------------------
SOURCE DEFINITION
---------------------------------------------------------------
5.0               Min magnitude associated with the run
-200 2009         First and last year in catalogue   (years)
  .2              Cell size                          (degrees)
  3               Smoothing radius                   (cells)
  0               Min. events for smooth             (count)
  0  50           Min and max depth                  (km)
999               Source depth                       (sdepth)     (0=sut,
999=auto,x=km)
---------------------------------------------------------------
PATH DEFINITION
---------------------------------------------------------------
1   0             Min. and max source-site distance km (0=auto)
0                 Short paths (ishortpaths) (0=elim,1=use rmin,2=adjust)
0                 File (.obs) with observation points (instead of default grid) (max
12 char.)
---------------------------------------------------------------
TIME SERIES
---------------------------------------------------------------
  1.               Peak frequency          (peakfr)     (1.0 or 10.0)
  1                Interpolation           (npint)      (0-9)
```

```
  1               Seismogram format      (iform)      (0=ASCII, 1=bin)
4096              Time series samples    (npts)       (4096)
  0               Time series length     (iall)       (0=truncated,1=complete)
  1               Type of motion         (itype)      (1=dis, 2=vel, 3=acc)
  0               Vertical component     (ivert)      (1=yes,0=no)
  1  90           Type of scaling        (iscale,iaz) (1=classic,2=pulsyn)(angle)
eurocode.dat      File with code response spectra for computing DGA (max 12 char.)
  0               Plot seismograms       (isis)       (1=yes,0=no)
```

You must specify the name of the run xxx (maximum 15 characters).

For the tsunami case some parameter is requested to assume specific values, in particular you have to choose:

- peak frequency 0.01 (tsunami lives at very long period!)
- interpolation 0
- seismogram format 0 (specific format for the core program syt2d.f)
- type of motion 1 (displacement, in particular vertical displacement, is the most important quantity to evaluate risk from tsunami)
- type of scaling 1 (the “extended source” case starting from pulsyn is not yet implemented in the classic run).

The case execution = 4 is not present in run for seismic waves, it allows to compute in automatic a single flow tsunamigrams for a grid of receivers from a single source or for a fixed number of sources.

To check if there are no problems in your input files, you can run the program makehaztsu.out changing the value of “Execution” first into 1 and then into 2. If everything is ok, you can run the program makehaztsu.out with the value 0.

Cells.par: parameter file for the first program fo the package (seismicity gridding)

```
parameters for program ecells   (filenames reading format: A40)
aaa.dat                         file with first earthquake catalogue
aaa.poc                         file with first polygons of validity
bbb.dat                         file with second earthquake catalogue
bbb.poc                         file with second polygons of validity
…
zzz.dat                         file with nth earthquake catalogue
zzz.poc                         file with nth polygons of validity
```

DETAILS

Each earthquake catalogue (.dat) has a polygon file (.poc) that defines its validity area. This is important mostly when dealing with neighboring countries, each one having its own earthquake catalogue available.

FAQS

Q: Do I have to prepare a .poc file if I don’t have to take care of neighboring countries?

A: Yes, this is required by the software. You can just assume the rectangle including all the events listed in your catalogue. Nevertheless, there is generally no need of considering events at distances larger than 200 km from the region for which you’ll compute the hazard.

WARNING

In general, when dealing with catalogues from neighboring countries, it's recommended to analyze and merge the earthquake catalogues with care about possible duplicates reported at different coordinates in neighboring catalogues.

aaa.dat earthquake catalogue with observed seismicity

```
1005 1 1 0 0 0 4347  1188  0520520  05200
1005 1 1 0 0 0 4150  1375  0520520  05200

…

1991 531 949 0 4428  1005  5320  0  0  00
1991 53110 4 0 4346  1296 10220  0  0  00
```

FORTRAN statement to read one record:

```
      read(1,1)iy,m,id,iho,mi,isc,la,lo,ih,m1,m2,
     *         m3,m4,idummy
    1 format(i4,5i2,i5,i6,5i3,i1)
```

EXPLANATION OF DATA:

```
iy     year
m      month
id     day
iho    hour
mi     minute
isc    second
la     latitude*100 (example:   1492 means  14.92 degrees North)
lo     longitude*100(example: -12372 means 123.72 degrees West)
ih     depth (km)
m1     magnitude1 *100
m2     magnitude2 *100
m3     magnitude3 *100
m4     magnitude4 *100
idummy unused, set to 0
```

DETAILS

This file is the catalogue of known events that affected the studied area in the past. In parallel, file .poc contains the polygon that defines the area for which the catalogue can be assumed valid. When running for adjacent countries you have to carefully investigate the completeness of each national earthquake catalogue in the areas where they overlap. In this case files .poc should be set so that each catalogue is used where it is best compared with the others.

FAQS

Q: For historical events I do not have all the information. What can I do?

A: Just include what you know. You must include at least latitude, longitude and magnitude obtained from intensity. Put a 0 in all other fields.

Q: My catalogue is not complete. What can I do?

A: Opposite to what happens with probabilistic methods, in this deterministic approach we don't care much about catalogue completeness. Here seismicity will be discretized into cells (0.2° x 0.2°), and only the biggest event that occurred in each cell will be considered. Furthermore, within seismogenic zones defined in file .pos, if magnitude observed in the catalogue is lower than 5 or no event is reported in the catalogue, magnitude 5 will be used by default.

Q: Can I include information on seismic potential of active faults known from geology, for which I have no information in the historical catalogue?

A: Yes, you can create a .dat file with seismic potential data rather than observations. But remember that the default run is meant to be made with observations! You can rerun programs with the catalogue of seismic potential, and consider it as a parametric test.

WARNING

If you know magnitude estimates in different scales (ie: Mb from body waves, MS from surface waves, Ml local magnitude, Mk macroseismic magnitude) you should usually store them separately in m1, m2, m3 and m4. Anyway, program cells will read just m1. So you should put in m1 the maximum magnitude value available. Program edcatsun.out can help you in this.

xxx.fps: file with list of available focal mechanisms

```
-------------------- FPGNDT.DAT -Revi:sed- July 1, 1977 ------------------------
NUMBEA YEARMODY HRMISEC LA.TITN LON.GITE DEPT MLMDMSMBMA AGEN AREADESCRI
NUMBEF ST1 D1 RA1  ST2 D2 RA2   PDI PI  TDI TI  BDI BI Q REFE AREADESCRI
NUMBEM M00 SF REFE DURA F2 M0XX ER1 M0YY ER2 M0ZZ ER3 M0XY ER4 M0XZ ER5 M0YZ ER6
NUMBET HDR SF M0 TVAL TD TAZ NVAL ND NAZ PVAL PD PAZ AST AD ARA BST BD BRA REFER
NUMBEU SF SMRR ER1 SMTT ER2 SMFF ER3 SMRT ER4 SMRF ER5 SMTF ER6
--------------------------------------------------------------------------------
   44A 19591223  929000 37.720N  14.610E  770  0 0 0 053      SICILY
00044F 077 43 004  344 87 132   041 29  289 34  161 43   0001 SICILY
   54A 19671031 2108000 37.840N  14.600E  380  0 0 0 050      SICILY
00054F 009 61 189  274 80 333   228 27  324 13  077 60   0001 SICILY
   57A 19680115  133000 37.890N  13.080E  200  0 0 0 051      SICILY
00057F 040 82 046  302 46 168   163 23  272 37  049 45   0001 SICILY
   58A 19680115  201000 37.780N  13.030E   30  0 0 0 054      SICILY
00058F 204 70 015  108 75 159   157 04  065 25  255 65   0001 SICILY
   58B 19680115  201085 37.750N  12.983E  100  0 0 054 0      SICILY
00058G 270 50 035  156 64 134   216 08  116 50  313 39   0002 SICILY

…

   88A 19760917  123000 38.100N  13.300E  330  0 0 0 044 XXXX SICILY
00088F 322 69 240  200 36 323   194 56  074 19  334 28   0001 SICILY
00676A 19771225 1150    37.000N  15.200E  090  0 0 0 0 0 XXXX ETNA
00676F 036 76 093  206 15 079   124 30  310 59  216 03   0038 ETNA
  601A 0000000000000    39.800N  19.000E  000  0 0 0 0 0 XXXX IONIAN SEA
  601F 030 90 180                                        0000 IONIAN SEA
```

FORTRAN statements to read the event records: (numer before label 'A',... - only for user)

```
      read(1,"(a)") rec1              ! character*80 rec1
  100 if(index('ABCDE',recl(6:6)).ne.0) read(rec1,2)
     *          labr,r1lat,lb,r1lon,lc,ide,ml,md,ms,mb,ma
      do ilab=1,6
         if(labr(ilab:ilab).eq.' ') labr(ilab:ilab)='0'
      enddo
      if(lb.eq.'S'.or.lb.eq.'s') r1lat=-r1lat
      if(lc.eq.'W'.or.lc.eq.'w') r1lon=-r1lon
      read(1,1) rec2                  ! character*80 rec2
      if(index('ABCDE',recl(6:6)).ne.0) then
           rec1=rec2
           go to 100
      endif
      if(index('FGHIL',recl(6:6)).ne.0)
     *     read(rec2,3) istr1,idip,irak1,istr2,idip2,irak2, iptre,ipplu,ittre,itplu
1 format(a80)
    2 format(a6,17x,f7.3,a1,f8.3,a1,1x,i4,1x,5i2)
    3 format(7x,2i3,1x,i3,2x,2i3,1x,i3,3x,2i3,2x,2i3)
```

EXPLANATION OF DATA

```
labr  a 6-character label (5 digits and 1 letter - A...M)
```

```
r1lat latitude in degrees (0≤r1lat≤90; example 45.68)
lb    latitude specification (N for North, S for South)
r1lon longitude in degrees (0≤r1lon≤180; example 120.88)
lc    longitude specification (E for East, W for West)
ide   depth in km multiplied * 10
mL    local magnitude multiplied * 10
md    duration magnitude multiplied * 10
ms    surface waves magnitude multiplied * 10
mb    body waves magnitude multiplied * 10
ma    other magnitude multiplied * 10
istr1 strike angle of plane A
idip1 dip angle of plane A
irak1 rake angle of plane A
istr2 strike angle of plane B
idip2 dip angle of plane B
irak2 rake angle of plane B
iptre p-axis trend
ipplu p-axis plunge
ittre t-axis trend
itplu t-axis plunge
```

DETAILS

This is the file with the list of focal mechanisms available for the region of interest. There are seven comment records to begin with (description of content for each type of record). After them, each fps has at least two records defined: record with label A (info for event) and F(fps). If you have different solutions proposed for the same event, you can enter them in the file with labels B and G (and then C and H, D and I and E and L) respectively. So at most 5 different solutions can be associated with the same event. All the solutions that you enter in the file will be used by the package, with no priorities. Records with label M (tensor components), T and U needn't to be defined.

You need to have at least one solution falling within each seismogenic zone defined in the file .pos. If you enter more than one mechanism per polygon program emecmed will compute a "representative" fps obtained as the average of the available moment tensors (each event and each solution together). This might not be a good choice in case the available solutions differ significantly one each other. It sounds more reasonable to include in the .FPS file just one representative focal mechanism per seismogenic zone.

FAQS

(none)

WARNING

The package will just read event index, coordinates, magnitude and strike, dip and rake of nodal plane A. For the sake of information completeness, you should better fill in all the other fields.

Event index (the numerical part of the label labr) must be different for different events. It must be equal only for different solutions associated with the same event.

Strike, dip and rake must be defined using the convention reported in Aki and Richards. Positive rake angles remain untouched. Negative rake angles should be transformed to positive performing the operation (360 + negative_rake). Here you have some examples:

Aki and Richards	Operation	Value for .fps file
90	none	90
-50	360-50	310
0	none	0
180	none	180
-160	360-160	200

xxx.poc: validity area for the earthquake catalogue

Polygon associated with the catalogue of seismicity (file xxx.dat)

```
xxxaaa0001
             6.0    36.0
            20.0    36.0
            20.0    48.0
             6.0    48.0
```

FORTRAN statement to skip the first record:

```
   read(1,*)
```

FORTRAN statements to read each polygon:

```
   read(1,1) label,num    ; polygon name
   read(1,*) xlon,ylat    ; vertex record
 1 format(a6,i4)
```

EXPLANATION OF DATA:

```
label a six-character label
num   a 4-digits number
xlon  logitude in degrees
ylat  latitude in degrees
```

DETAILS

This file defines the area where the earthquake catalogue (xxx.dat) has to be used. When running for a single nation the area should cover the whole territory plus about 100 km around the political borders. When running for adjacent nations one .poc file has to be defined for each catalogue, and the areas defined by each .poc file must not overlap. For each catalogue, only events falling within the relative .poc file will be considered.

FAQS

Q: Should I follow any specific order in the definition of the coordinates?

A: Yes, for each polygon corners should be listed following either clockwise or counterclockwise order, starting from any point.

Q: Should I close each polygon defining the same corner at the beginning and at the end of the coordinate list?

A: No.

WARNING

Please follow strictly the format of the example file. Do not use TABS, just SPACES (and count them well…) when editing the files. With some editors tabs are inserted automatically to facilitate indentation. Be sure to avoid that option! Programs will complain quite a lot…

Only one polygon has to be defined in file .poc, associated with the corresponding earthquake catalogue (.dat)

xxx.por areas associated with different structural models

Polygons associated with regional structural models (files *.spl and *.spr and *.spt)

```
struct0001
            11.750  44.250
            11.900  44.600
            11.300  44.650
            11.200  44.380
struct0002
            11.200  44.380
```

```
            11.300  44.650
            10.900  44.700
            10.750  44.500
struct0003
            10.750  44.500
            10.600  44.300
            11.500  44.000
            11.750  44.250
            11.200  44.380
```

FORTRAN statement to skip the first record:

```
   read(1,*)
```

FORTRAN statements to read each polygon:

```
   read(1,1) label,num    ; polygon name
   read(1,*) xlon,ylat    ; vertex record
 1 format(a6,i4)
```

EXPLANATION OF DATA:

```
label a six-character label
num   a 4-digits number
xlon  logitude in degrees
ylat  latitude in degrees
```

DETAILS

This file defines the areas associated with different regional structural models, and therefore with files .spt. For minimization of CPU time, synthetic seismograms will be computed in a loop over structural models: there will be one .isg file for each polygon defined in the .por file.

The whole territory must be covered by the polygons, since synthetic seismograms will only be computed within them. Polygons should not overlap, should not be defined off the coasts and should follow roughly the political borders in order to facilitate in the future the computation of joint maps with adjacent nations.

FAQS

Q: Should I follow any specific order in the definition of the coordinates?

A: Yes, for each polygon corners should be listed following either clockwise or counterclockwise order, starting from any point.

Q: Should I close each polygon defining the same corner at the beginning and at the end of the coordinate list?

A: No.

WARNING

- It is strongly suggested that for .por files polygon sides defined exactly in the NS and EW directions are given coordinates ending with decimals .1, .3, .5, .7 or .9 when the cell size is (as always recommended) 0.2 degrees.
- The label associated with each polygon must be made of 6 chars and 3 digits.
- Please follow strictly the format of the example file. Do not use TABS, just SPACES (and count them well…) when editing the files. With some editors tabs are inserted automatically to facilitate indentation. Be sure to avoid that option! Programs will complain quite a lot…

- It is not necessary to set numbers of polygons (num) sequentially, but the number must be unique. It is very convenient that uniqueness remain even when joining the results of different countries. So please ask about numbers that you could better use.

xxx.pos: seismogenic zones

Seismogenic zones. In file xxx.fps at least one mechanism per polygon!

```
region0001 6.14
            11.200  44.600
            11.600  44.600
            11.600  44.400
            11.200  44.400
region0002 6.60
            10.800  44.200
            11.200  44.200
            11.200  44.000
            10.800  44.000
```

FORTRAN statement to skip the first record:

```
   read(1,*)
```

FORTRAN statements to read each polygon:

```
   read(1,1) label,num    ; polygon name
   read(1,*) xlon,ylat    ; vertex record
 1 format(a6,i4,f4.2)
```

EXPLANATION OF DATA:

```
label a six-character label
num   a 4-digits number
lmmax  maximum magnitude for seismogenic sources
x     logitude in degrees
y     latitude in degrees
```

DETAILS

This file defines the geometry of the seismogenic zones. In the computation of synthetic seismograms, sources will be considered only within these polygons.

Polygons should not overlap, and should account for all the active areas that could affect the region where seismograms will be computed, even off the coast and outside the political borders. Therefore they can (and should…) be placed also outside the area covered by .por file.

Later programs eselmec and emecmed will assign one single focal mechanism to the sources belonging to the same seismogenic zone.

The magnitude for seismogenic zones is used only for accepting the zones. If the maximum magnitude is greater than the threshold alarm of CN and M8 region, the seismogenic zone is retained. If the lmmax value is blank the seismogenic zone is retained.

FAQS

Q: Should I follow any specific order in the definition of the coordinates?

A: Yes, for each polygon corners should be listed following either clockwise or counterclockwise order, starting from any point.

Q: Should I close each polygon, defining the same corner at the beginning and at the end of the coordinate list?

A: No.

WARNING

- It is strongly suggested that for .pos files polygon sides defined exactly in the NS and EW directions are given coordinates ending with decimals .0, .2, .4, .6 or .8 when the cell size is (as always recommended) 0.2 degrees.
- The label associated with each polygon must be made of 6 chars and 4 digits.
- Please follow strictly the format of the example file. Do not use TABS, just SPACES (and count them well…) when editing the files. With some editors tabs are inserted automatically to facilitate indentation. Be sure to avoid that option! Programs will complain quite a lot…
- It is not necessary to set numbers of polygons (num) sequentially, but the number must be unique. It is very convenient that uniqueness remain even when joining the results of different countries. So please ask about numbers that you could better use.

File xxx.sut: source information

If execution parameter in file makehaz.par is set to 3 or 4 you need to write for your own (hopefully using information from literature) file xxx.sut that contains source parameters. This file has the following form.

```
 “    label          slon      slat    depth   strike   dip   rake      mag  isub
weight      tshift MECM16 prova.mec    prova.mag

 0904aa00110028   14.4000  43.4000  10.000 145.000  76.000 180.000  8.50    1
0.1000E+01  0.0000E+00

 0904aa00110029   15.5000  42.2000  8.000  144.000  90.000 180.000  8.00    1
0.1000E+01  0.0000E+00
```

DETAILS

The names of the the different sources have to be indicated by string of 14 alphanumerical characters.

File xxx.obs: receivers information

```
      lon      Lat      struc    rdep
   109.2925 11.0014   0011    0.000
   109.2041 12.2443   0011    0.000
   109.0962 10.3446   0012    0.000
   119.5907 15.9000   0012    0.000
```

File with spectra (xxx0001.spt...xxx000n.spt): tsunami spectra associated with structural model

DETAILS

These are files generated by program tsu, which is part of the modal package developed at the University of Trieste. This program read structural files with ocean layers (*.stp files, see manual of modal summation package) and file tsu.par (analogous to ray.par or lov.par) and then compute spectra files. In 1D modeling one has to use the spectrum corresponding to the structure at the receiver.

gusev01.xy: curves for scaling of synthetic seismograms

```
gusev21
-3.658  21
-3.465  21
-3.162  21
-2.826  21
-2.445  21
-2.138  20.992
-1.672  20.984
-1.209  20.97
-0.824  20.977
-0.407  20.967
-0.163  20.936
0.071   20.851
0.213   20.761

…

1.936   16.929
2.011   16.732
2.109   16.498
2.202   16.259
2.287   16.032
2.365   15.832
```

DETAILS

These files define the frequency scaling laws used by program efft to scale the synthetic seismograms with magnitude. Programs nsgl, nsgr, nsgv generate the seismograms in the point source approximation with M0 = 1E+20 dyn cm. The true scalar seismic moment is obtained with the rule

$$\log(M0) = 1.5\ M + 16.05$$

where M is the magnitude. For the scaling, each seismogram is Fourier-transformed and its amplitude spectrum is multiplied by the curve having the right M0, obtained by bilinear interpolation from the basic curves reported by Aki and based on the original paper by Gusev. With another Fourier transform the signals are again defined in the time domain.

FAQS

Q: Can I use different scaling laws?

A: Yes, efft program is already set for using ω2 laws. You can also define different curves in the gusev??.xy files, but at this moment you can not use different filenames.

WARNING

- Please note that with this kind of scaling source dimension is taken into account, but not the details of the rupturing process. Therefore the shape of the unscaled signal will be affected by thew scaling, but not its duration. To account for duration, complex sources should be used, built as a superposition of point sources properly weighted and distributed in space and time. This is beyond the scope of this package. Please refer to the work by Angela Saraò et al.
- The Gusev scaling laws are appropriate and tested against observations for shallow events (roughly speaking h < 20 km). For deep events different laws should be used, and thoroughly tested.
- If you prepare new digitized curves, you have to define them for the log(M0) values of 21,22,… 30 as in the original gusev??.xy files, maintaining the same filenames. No more than 99 samples should be defined. Sampling interval doesn't need to be regular.

tsugmt.sh: job for plotting maps

This is a gmt script for plotting maps, the command you have to type is

```
tsugmt.sh [-options] <input file>
```

The option must be chosen according to the extensions of the <input file>.

The command generates two output files (one in pdf and one in ps format). The content is the same.

The option are listed in the table below.

Options	Extension of input files	Type of output data	Extension of output files
	.tmx	Vertical displacement	.pdf
4	t.tmx	Arrival time of the maximum for each receiver	.t.pdf
b	all extensions	maps in gray scale (Black and white)	.bw.pdf
c	all extensions	no title (Clear)	.pdf
t1 t2 t5	all extensions	maps with Topography with different level of detail (t1 for regional scale (1km) – t2 for intermediate scale (etopo2)- t5 for global scale (etopo5))	.pdf

Description of the maps.

xxx.gmt.pdf focal mechanisms (within seismogenic zones)

xxx.poc.pdf polygon associated with earthquake catalogue

xxx.por.pdf regional polygons associated with structural models

xxx.pos.pdf polygons associated with seismogenic zones

xxx.ung.pdf gridded magnitudes (within earthquake catalogue polygons)

xxx.uni.pdf smoothed magnitudes (within seismogenic zones)

xxx.unm.pdf smoothed magnitudes (within earthquake catalogue polygons)

xxx.uns.pdf smoothed final magnitudes (within seismogenic zones)

xxxf0tsz.amx.pdf vertical displacement peak values (inside regional polygons)

xxxf0tsz.amx.t.pdf arrival time of the peak

Note. The minimum found in file.cou and file.fin (from which plotted peaks are taken) is not the absolute minimum but is the minimum arriving before the maximum (and in someway announcing it).

In program couanim.f you can also choice to extract the lapse time between this minimun and the maximum instead of the arrival time of the maximum.

Extended source run

Besides the classic run with the option showed in makehaz.par this routine offers other possibilities, mainly “extended source run” and “animation run”. In these cases, at the state of the art, you can't launch the script haztsu but you have to run the various programs in sequence separately and, in particular, the program PULSYN has to be used. The program syt2d.f reads the input from file xxx0001.isg, that contains information about source data and source-receiver paths. In the classic run (point sources) the .isg file is generated by program tsuinput.f and its header is something like this

```
“I N P U T   P A R A M E T .   F O R     246  S E I S M O G R A M (S)”.
```

In the case of extended source run, the .isg file is generated by program PULSYN and its header is different, in fact the number written in it, that in this case indicates the number of subsources, follows the form “npair=”, e.g.

```
“EXTENDED SOURCE               npair= 246  generated by PULSYN    ” .
```

When “syt2d.f” reads “npair=” it works in extended source mode and instead of writing one seismogram for each source-receiver path, it computes the weighted seismograms for each subsource and then it sums them before writing output.

List of required input files.

To execute this run you also different input files from classic run (but makehaz.par is still needed to define some general parameters in the program flow), in particular you need:

Specific for pulsyn06.f

- pul.par: generic input file that indicates source and receiver files
- xxxxxx.src: file with source parameters
- xxxxxx.ste: file with receivers position and relative structure/spectra labels
- GUSEV83.TB5: file with Gusev’s curves

Spectra

- xxxxxx0001.spt
- ‘’
- ‘’
- ‘’
- xxxxxxx000n.spt

(Pulsyn requires that spectra labels are composed by 6 letters/numbers, corresponding to the run’s name, and 4 numbers indicating the spectrum relative to a specific receiver. For more information about pulsyn and specific input files you should read pulsyn manual)

Input to compute seismograms

- syt.par: Input file for program syt2d.f, in normal run is prepared by program tsuinput.f, in this case you have to write it. It has the following form

```
“Parameters for program syt generated by tsuinput: prova
0    0=ASCII  1=BINARY seismogram files
xxx000n.spt                                structure         1
xxx000n.isg                                input parameters  1
xxx000m.spt                                structure         2
xxx000m.isg                                input parameters  2
xxx000k.spt                                structure         3
xxx000k.isg                                input parameters  3
```

Sequence of programs to be launched

pulsyn06.out

Prepares input to compute tsunamigrams for extended source case.

syt2d.out

Compute tsunamigrams for extended source for one or more receiver positions.

efft.out

Scaling and filtering of the signals (ask Franco for details). VERY IMPORTANT: before launch it you have to modify the parameter file “fft.par” generated by haztsu considering not weighted seismograms for point-source case (remember that the signal from extended source is already weighted for the chosen magnitude). In particular you have to set to 0 the parameter iscale.

```
SCALING LAW ----------------------------------------------------------------
 0         iscale  index (0 no, 1 Gusev, 2 Gusev at freq. gufreq, 3 f**2)
```

couanim.out

Find, for each seismogram, the maximum of vertical displacement, the minimum, the interval time between the two quantities and the arrival time of the minimum.

tsufinmax.out

Chooses the maximum peak for every receiver and generates output named xxxf0tsz.tmx ready to be plotted by gmt.

tsumaxsig.out

Selects, for every receiver, the complete seismogram containing the peak values.

*sh tsugmt.sh *.tmx*

Launch the script for plotting (others script options are the same of the default run).

Run for animation (not implemented on is01)

The basic idea of this kind of run is to use an option in program couanim.f that, instead of the maximum in a file.cou, allows to create a file.anx for every temporal sample. Thus, considering a grid of receivers, you can have a series of images representing the motion in the entire basin developing in time, and with specific program animate them. Also for this type of run for the moment you can't use the script haztsu but you have to run programs separately.

List of required input files

To execute this run you need different input file than classic run, in particular you need:

angmt.sh

Is the script for plotting images that collected together will create the animation

File with source input and information for path computation

In this case you have to produce by yourself the file patgen.par, the correct form follows.

```
“Parameters for program epatgen generated by makehaz:  xxx
xxx.sut                                File with selected sources

xxx.por                                File with structural polygons

0                                        File with  observation coord (0=grid)

  1.                                     Min. source site-distance km (0=auto)

500.                                     Max. source-site distance km (0=auto)

1                                        Short paths: 0=elim,1=use rmin,2=adjust”
```

DETAILS

The minimum source-site distance has been chosen to be 1, this is because syt2d.f has no computational problems for distance greater than zero; this doesn't mean that results at short distances are good from a physical point of view. The important thing is to put a value different from 0 to avoid numerical problems that will block the efft.out.

Then, the default files are needed:

xxx0001.spt: file with spectrum associated with polygon 0001 defined in xxx.por
.
.
.
xxx000n.spt: the file with the spectrum associated with polygon 000n in xxx.por
tsugmt.sh: script for plotting
gusev01.xy: Spectral curves for scaling seismograms with magnitude

Sequence of programs to be launched

tsupatgen.out

Generates source-receiver path

tsuinput.out

generates file.isg and parameter file syt.par

syt2d.out

Compute tsunamigrams for source receivers pairs.

efft.out

Scaling seismograms

couanim.out

Is the program that creates the temporal series of files necessary for the animation.

IMPORTANT: before launch couanim.out you have to modify the parameter file cou.par. This file is created by program syt2d.out and in its default form it's not set for animation run. Its default form is the following:

```
“Parameters for program cou generated by tsu:       xxx
 0    0=ASCII  1=BINARY seismogram files
 0    1=use frequency information  0=do not
#
prova0014f0.tsz
prova0012f0.tsz
prova0016f0.tsz
prova0252f0.tsz
”
```

Where xxx is again root name and n,m,l,k are index of structures.

If in the fourth line there is # couanim would work in the usual way; to work in animation mode you have to put in place of # the index of source (of sources if you're interested in animate more than one source signal)preceded by symbol *

For example if you want animation for two sources you write

```
“Parameters for program cou generated by tsu:       prova
 0    0=ASCII  1=BINARY seismogram files
 0    1=use frequency information  0=do not
*sour name 1
*sour name 2
prova0014f0.tsz
prova0012f0.tsz
prova0016f0.tsz
prova0252f0.tsz
”
```

Inside the code there is a command that launch the script for plotting ps for every temporal sample(and for every source if there are more than one) without saving every file.anx, and then from every file.ps produces (using program raster) file.jpg. The names of the .jpeg file are sournsamplexxxxf0tsz.anx.jpg where n is the source index found in couanim.par and sample goes from 1 to 4096.

WARNING: next step has to be implemented on is01

```
mencoder 'mf://*.jpg' -mf w=800:h=600:fps=5:type=jpg -ovc lavc -lavcopts
vcodec=mpeg4:mbd=2:trell -oac copy -o anim.avi
```

Is the command necessary for run the program mencoder that from temporal series of jpg files creates animation (anim.avi). The parameter fps is the number of frames per seconds and set the speed and the duration of the animation

Input and example of run

You can find input data for using this suite at
/XDST/Examples/Tsunami/2D/Point/Input,
an example of classic run at
/XDST/Examples/Tsunami/2D/Point,
input data for using the extended suite at
/XDST/Examples/Tsunami/2D/Extended/Input,
an example of extended source run at
/XDST/Examples/Tsunami/2D/Extended.