UNIVERSITÀ
DEGLI STUDI DI TRIESTE

DMG dipartimento di matematica e geoscienze

The DMG Manuals

Genetic algorithm

Computation of error on moment tensor solutions by genetic algorithm

Document name:	DMGGenetic.pdf
Document Version:	1.0
Date Issued:	3 October 2017 15:24
Author:	Franco Vaccari

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

Table of Contents

Computation of error on moment tensor solutions by genetic algorithm

Input

The input files required to run genetic algorithm are the output from INPAR inversion, **moment.out**, the polarity file, named **pol.dat**, and the parameter file **listvar.inp.**
Below an example of the file with the polarities (pol.dat):

```
5
242.6 77.8 +1
265.7 51.6 +1
237.8 58.9 +1
255.9 30.1 +1
283.8 54.7 +1
```

These file must be placed in the working directory by the user.
Other input files necessary to run the programs are automatically generated running the program **inpgene_is** (source code: inpgene_is.f). The program generates input files for the first factorization (**inp14**), for the second factorization (**inp17**), for computing the contour lines (**inpiso**) and for plotting (**inpsurf**). Some of the input files requires few hand- changes to introduce parameters known only after that the first factorization has been done.

First factorization

Run the program **sigma15_is**. This program uses the input **moment.out** to compute joint model and noise standard deviation. These are written in the output file **sigmat.out**. The file moment.new is also created.
Then run the program for the first factorization **sga14_is** (source code: sga14_is.f).

The program requires an input file of parameters, named **inp14**. Below is an example of input file with commonly used values:

```
201 population number
100 number of generations
1. crossover probability
.001 mutation probability
1. interchange probability
2. number of copies of the best string
1 number of subintervals
moment.out input data
out14. SGA output
9 number of channels
```

⚠➡ Warning: the file must finish with an empty row! ⬅⚠

The program will automatically generate 20 output files **out14.#** (where # = 1, 2, . . .20). with different correlation to polarities. One of these will be used in the second factorization,automatically selected by the program or alternatively user selected.

Second factorization

Run the program for second factorization **sga17_is** (source code: sga17_is.f). To run the program you need the input file **inp17** and the output file selected after first factorization **out14.#**. Check the input file in order to be sure that the **out14.#** is properly selected. Check that the interval for the scalar moment M0 is an appropriate one. Below an example of **inp17** file:

```
201 population number
100 number of generations
1. crossover probability
.001 mutation probability
1. interchange probability
2. number of copies of the best string
1 number of subintervals
moment.out input data file
out17 SGA output for second factorization
9 number of channels
3 correlation of signs demanded
out14.1 SGA output of first factorization
stf.mem STF memory
rj.mem RJ plot memory
fps.mem FPS plot memory
m0.mem M0 memory
conf.tab confidence values tab.
.0 .999 pl mi, pl ma
.472189E+04 stfm(1,...,nsub)
-1 .001 .006 pm0[%], m0(min), m0(max)
50 ns2 (stfm sampling)
16 ns1 (dstf sampling)
100 100 nxm1, nym1 (RJ memory matrix size)
100 100 nxm2, nym2 (FPS memory matrix size)
100 nsm (M0 memory vector size)
```

The first ten rows are identical to those of the input file of the first factorization, except for the output filename at the ninth row. After the first ten lines, the parameters are defined as follows:

- correlation: it must be specified an integer value between 1 and 20; in such a way you specify the minimum correlation value required.
  The twenty values are fixed in the factorization program as follows:
  (1) 0.9; (2) 0.8; (3) 0.7, (4) 0.6, (5) 0.5, (6) 0.4, (7) 0.3, (8) 0.2, (9) 0.1, (10) 0.0, (11) -0.1, (12) -0.2, (13) -0.3, (14) -0.4, (15) -0.5, (16) -0.6, (17) -0.7, (18) -0.8, (19) -0.9, (20) - 1.1.
- output of the first factorization;
- matrix for the Source Time Function (STF);
- matrix for the resolution in the Riedesel - Jordan (RJ) formalism; • matrix for the fault plane solutions;
- matrix for the scalar moment;
- table with the confidence values;
- keep these values fixed at .0 .999;
- maximum value of the STF in the output of the first factorization (row 18 in the output file of the first factorization); actually you must insert a value greater of the 20%;
- specify the interval of the scalar seismic moment. You have two possibilities: specify the first number ”pm” positive, then the values between M0(1 – pm/100) and M0 (1 + pm/ 100), where M0 is the seismic moment corresponding to the best solution, are considered. In such case the two following numbers are not used. If ”pm” is taken as a negative number, the two following numbers are the maximum and minimum value of the seismic moment interval;
- ns2: number of rows (y coordinate) of the STF matrix;
- ns1: subinterval number for the x axis for the STF matrix; the number of column is given by (number of triangles +1) × ns1;
- columns × rows matrix RJ solution;
- columns × rows matrix fault plane solutions;
- array dimension for the scalar moment;

⚠➡ Warning: the file must finish with an empty row! ⬅⚠

Computation of the contour lines

To compute contour lines corresponding to a certain confidence level run the program **genetic2gmt05_is** (source code: genetic2gmt05_is.f). Input files **inpsurf** and **inpiso** are needed. In the input file **inpiso** you can add other confidence values; as default only three values are used: 68.30%, 90.0%, 95.40%.

Below an example of the **inpiso** file:

```
stf.ppd    output with the contour line values
stf.mem    STF memory matrix
200 50     columns × rows of the STF matrix
68.30 first confidence value
90.0  second confidence value
95.40 third confidence value
-1    negative integer to indicate the END of file
```

⚠➡ Warning: the file must finish with an empty row! ⬅⚠

Below an example of the **inpsurf** file:

```
stf.mem
stf.dat
200
50
1 scale factor of ppd values divided by his reciprocal fps.mem
fps.dat
60
60
1
rj.mem
rj.dat
60
60
1
m0.mem
m0.dat
60
1
1
```

The computation of the contour levels must be done only for the STF, the values of the isolines should be used for all the other solutions (mechanism and scalar moment). An example of output file **stf.ppd** is given below:

```
stf.mem
Confidence level 68.30% ==> PPD isoline z= .895059E-31
( 24 iterations needed)
Confidence level 90.00% ==> PPD isoline z= .252355E-31
( 21 iterations needed)
Confidence level 95.40% ==> PPD isoline z= .109260E-31
( 23 iterations needed)
```

Notice: in the case that, after running **genetic2gmt05_is**, you get isolines equal to 0, this means that confidence areas are too small to be plotted by GMT. In such a case you should repeat INPAR inversion adding one more triangle, in order to have less restrictive parameterization. Then repeat genetic algorithm computation by section 1.

The program will create all the input file for GMT plot, that are **devia.xy** (the file of deviatoric plane), **rj.xy** (coordinates and symbols of RJ solution), **fps.xy** (fault plane solution coordinates and symbols), **stf.xy** (STF coordinates and symbols), **m0.xy** (seismic moment coordinates and symbols).

A file named **solutions.txt** will be also automatically created with the solution of the genetic algorithm inversion.
The GMT script **plsgen** is automatically created.

Plotting by GMT

Run the script **plsgen** (after changing it to executable using shell command:

```
chmod +x plsgen
```

A graphic file named **solgen.ps** will be created with the plot of solution.

References

Sílený, J., Earthquake source parameters and their confidence regions by a genetic algorithm with a “memory”, Geophys. J. Int., 1998, 134, 228-242.