Simple model of landslide failure surface and displacement

Developing a python interface to estimate the SLBL surface and the displacement according to the model, and comparing to field data

Léo Letellier - intern
UNIL – Risk Group
08/2023

SFA - Setup

- Install python using miniconda
  - Run install_miniconda.bat
- Configure a virtual environment installing needed packages
  - Run env_setup.bat
- Start the python interface
  - Run SFA.bat

All the process is automated within the bat files, all that is needed is to launch them one by one and wait that the console window disappear before launching the next one. Once the virtual environment is ready, you just need to launch SFA when you need it, no need to modify again the environment.

Warning:

- The graphs displayed by this interface are powered by matplotlib. The majority of them are displayed with automatic scale. Thus the scale is often different for the axis. With different scales, the displayed angles are actually deformed. Consider manual adjustment in the graph parameters if you need angle accuracy for display purpose. A checkable button next to the graph button allows to set the same scale for each axis if checked.

Preparing the data

- All points must be sorted according to the x-axis.
- Be wary that python use a dot decimal separator (« 3.1415 » instead of « 3,1415 »).
- The column order in the loaded files is important, please refer to the README file.

Modeling the failure surface : SLBL approach

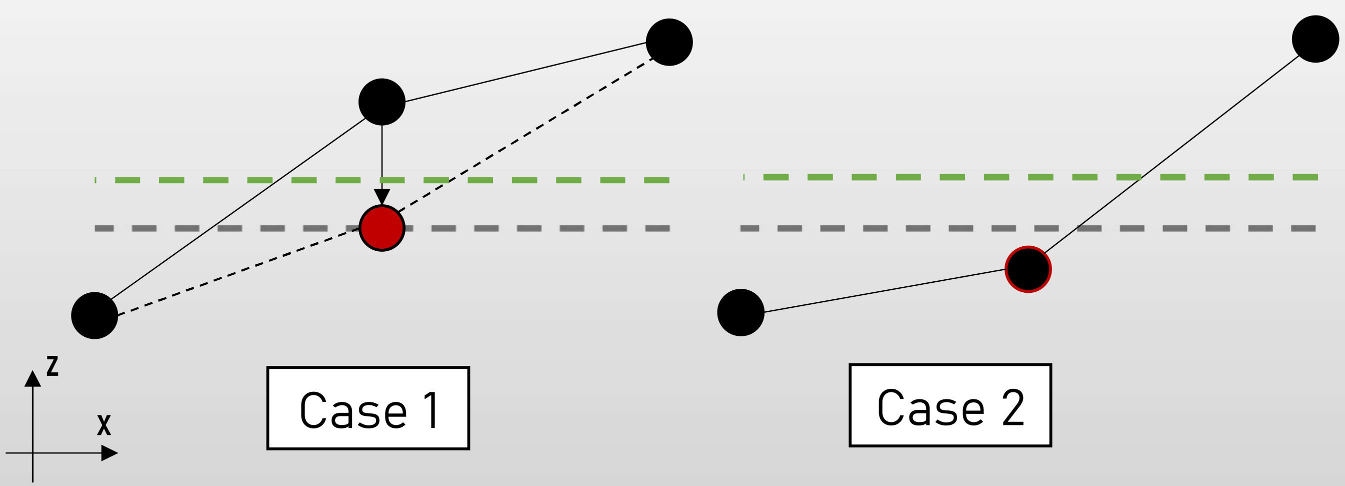
Overview of the method:

Toward preliminary hazard assessment using DEM topographic analysis and simple mechanic modelling by mean of the sloping local base level, Jaboyedoff et al., 2004

Iterative algorithm « Minimum wise » :

- $slbl_{temp[i]} = \frac{slbl_{-1}[i-1]+slbl_{-1}[i+1]}{2} - C$
- $slbl[i] = \min(slbl_{temp[i]}, slbl_{-1}[i])$

C is a tolerance parameter relative to the landslide geometric features



Minimum is temp value

Minimum is current value

Matrix expression: $\boxed{M.Z = C}$

$$\mathrm{M} = \begin{bmatrix} 1 & -0.5 & \cdots & & 0 \\ -0.5 & 1 & & & \\ \vdots & & \ddots & & \vdots \\ & & & 1 & -0.5 \\ 0 & & \cdots & -0.5 & 1 \end{bmatrix}$$

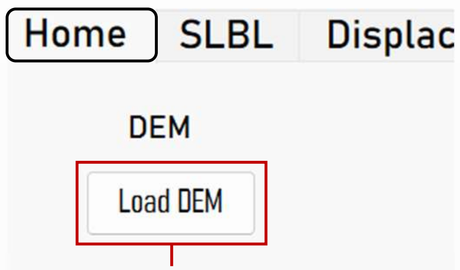
$$Z = \begin{bmatrix} z_1 \\ \vdots \\ z_{n-1} \end{bmatrix}$$

$$\mathrm{C} = \begin{bmatrix} z_0 - c \\ -c \\ \vdots \\ -c \\ z_n - c \end{bmatrix}$$

Modeling the failure surface : SLBL approach

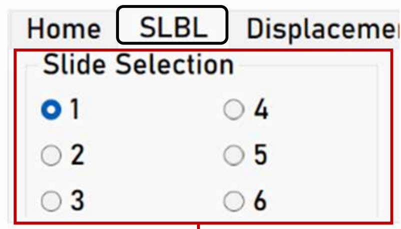
Implementation in the python interface:

1. Loading the DEM along the profile



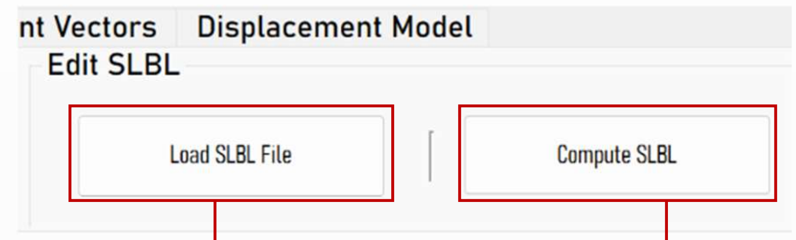
The SLBL data it computed according to the points of the DEM.

2. Selecting the memory slot to edit



Six SLBL can be stored at the same time.

3. SLBL can be loaded from a file or computed



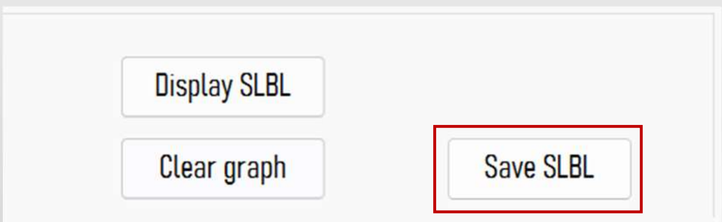
The file formats are specified in the README.md.

Opens the window for computation.

4. Display the computed SLBL

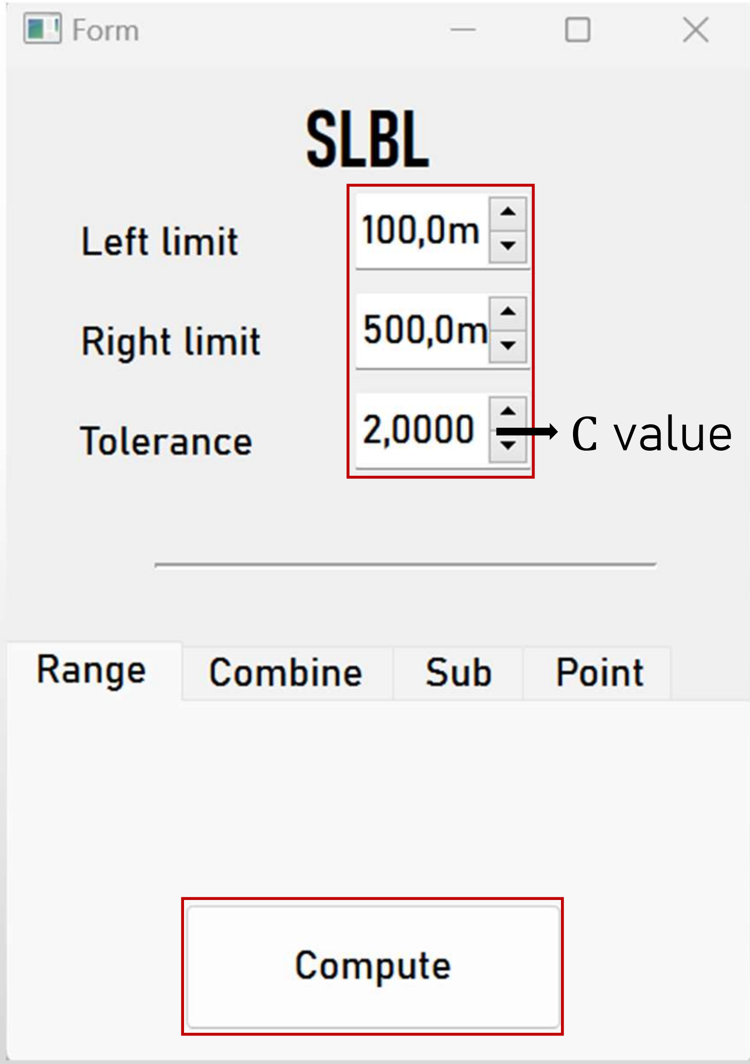
Display the SLBL of the selected slot along with the DEM.

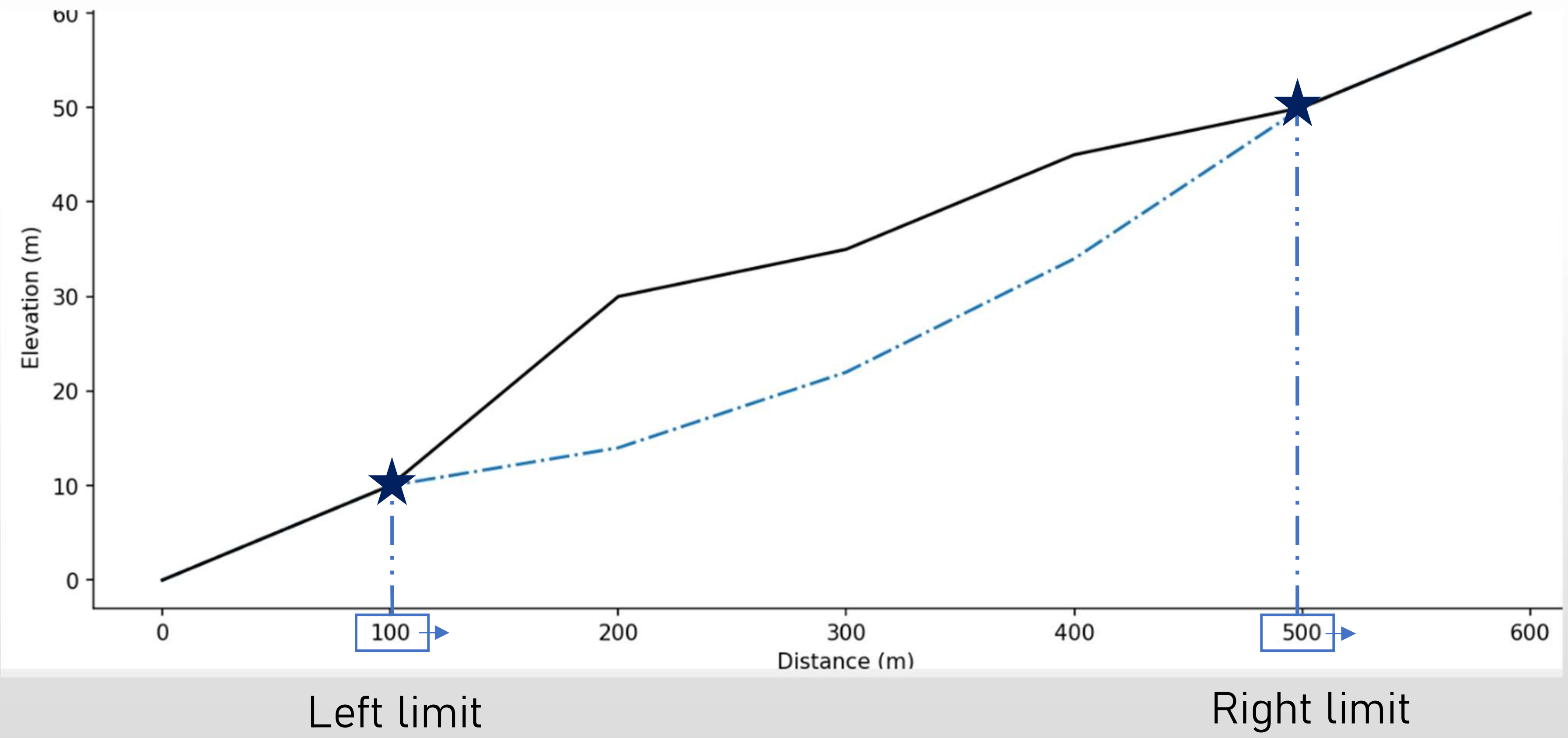
5. Export the computed SLBL for external use



Modeling the failure surface : SLBL approach

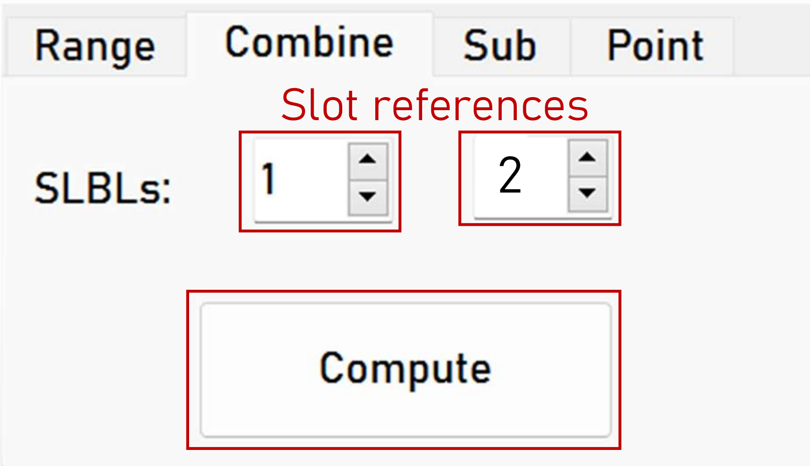
Compute SLBL with known range within the cross-section:





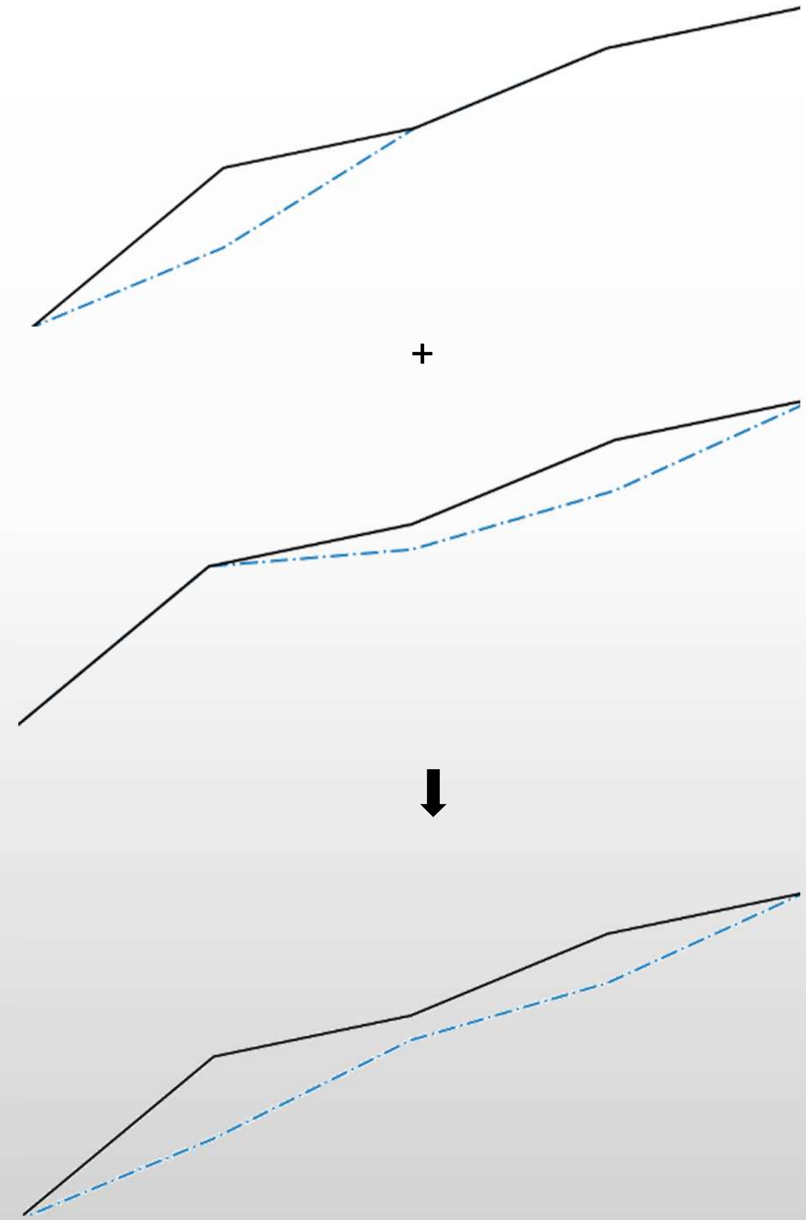
Modeling the failure surface : SLBL approach

Compute SLBL using two existing features:



Content of slot 1:

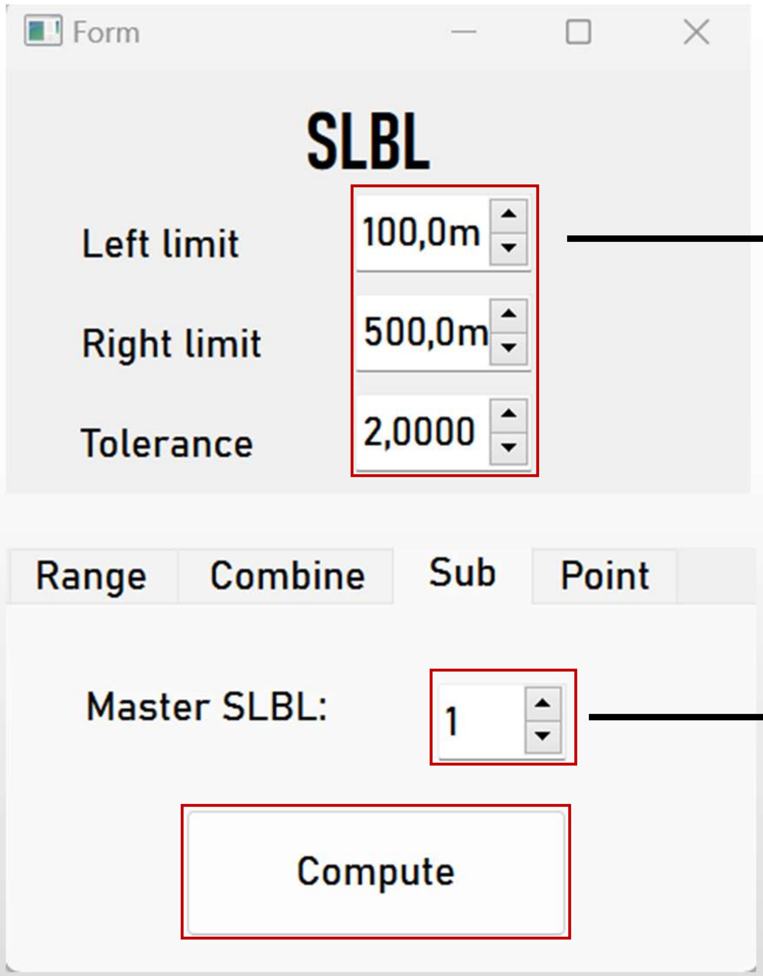
Content of slot 2:



Minimum of two features (SLBL)

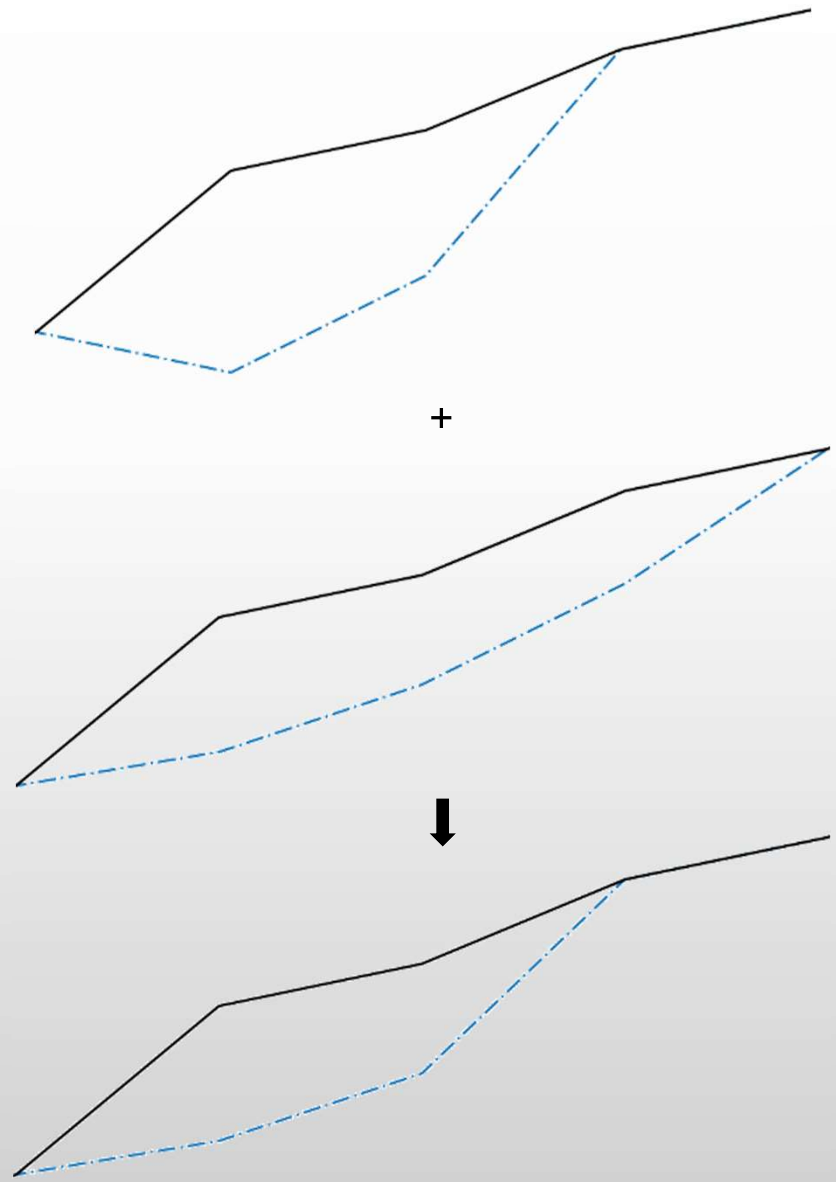
Modeling the failure surface : SLBL approach

Compute SLBL using one existing features:



Computed as range

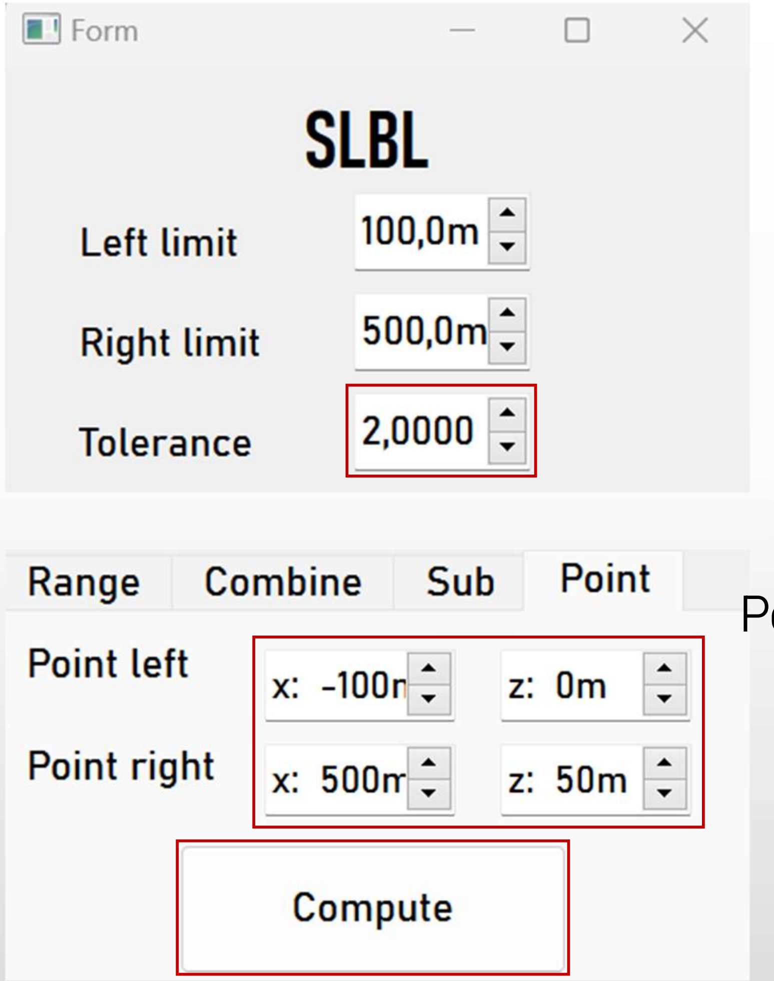
Content of master

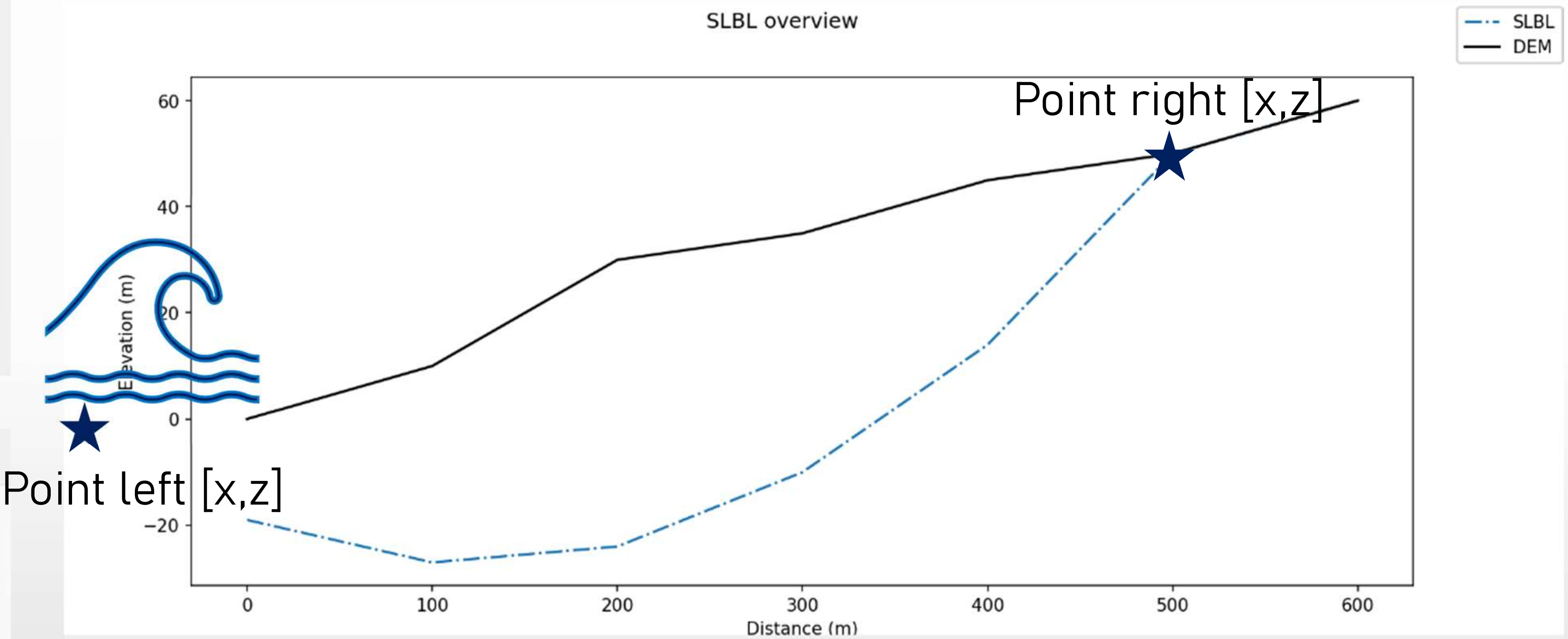


Maximum of two features (SLBL, faults, …)

Modeling the failure surface : SLBL approach

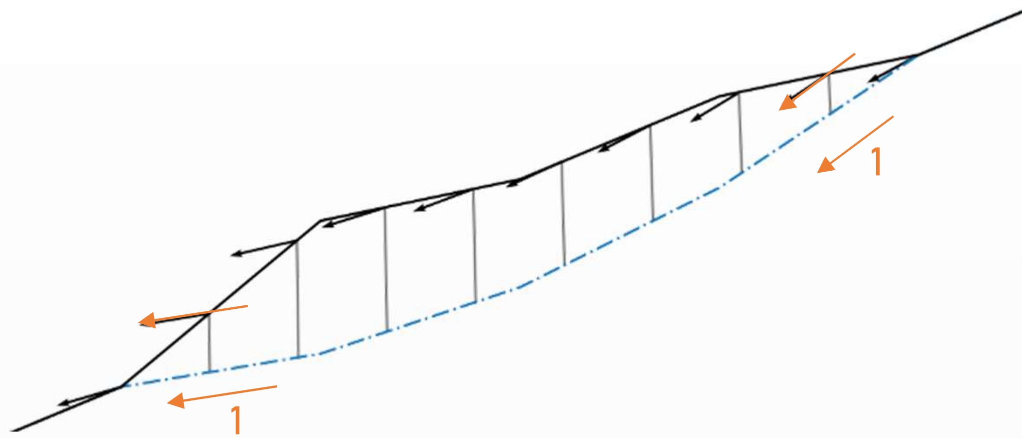
Compute SLBL using two points (inside or outside cross-section):





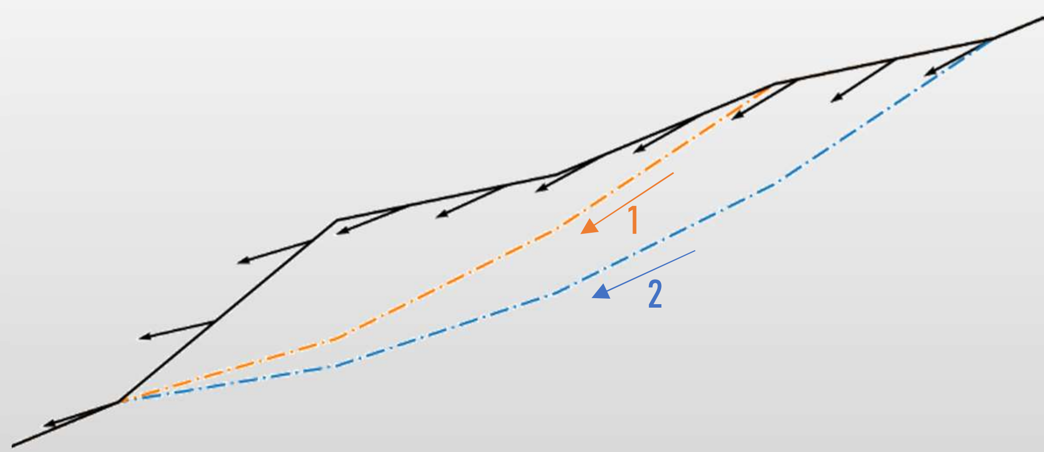
Model of displacement

- Basic displacement:
  - Following the failure surface (orientation of the vector)
  - Movement is conserved along the section (normalized vectors)
  - Movement is the same on one slide of the landslide (perpendicular to the local slope)



Single SLBLs with normal vectors

- Multi-layered failure surface:
  - Global displacement is the sum of weighted basic displacement for each failure surface

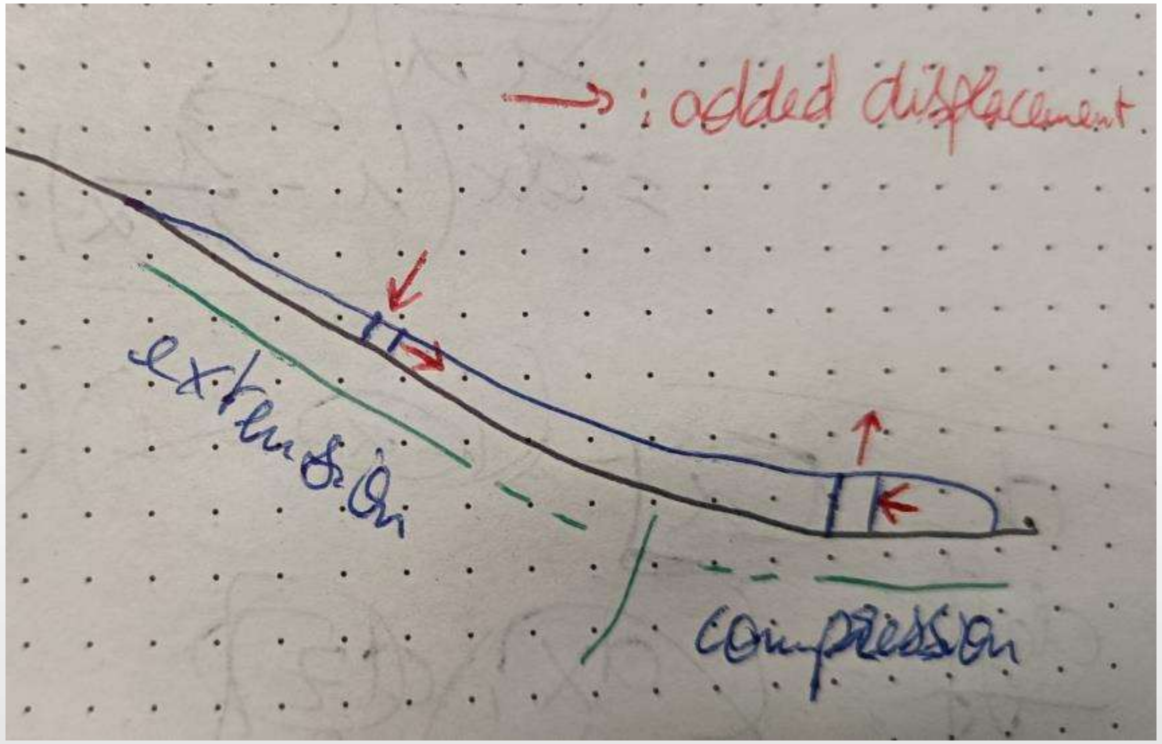


Multiple SLBLs with weights

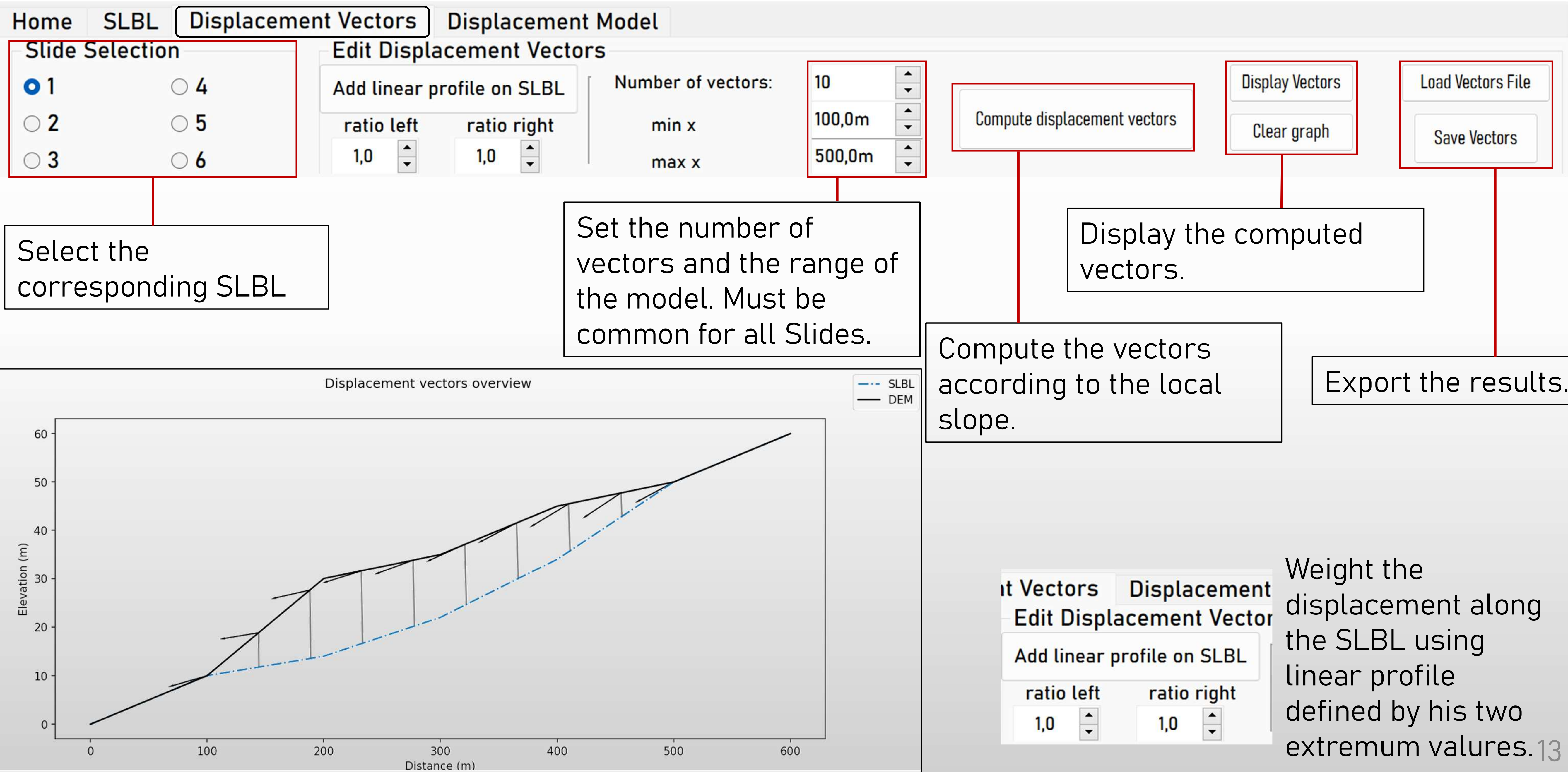
Model of displacement

NOT IMPLEMENTED

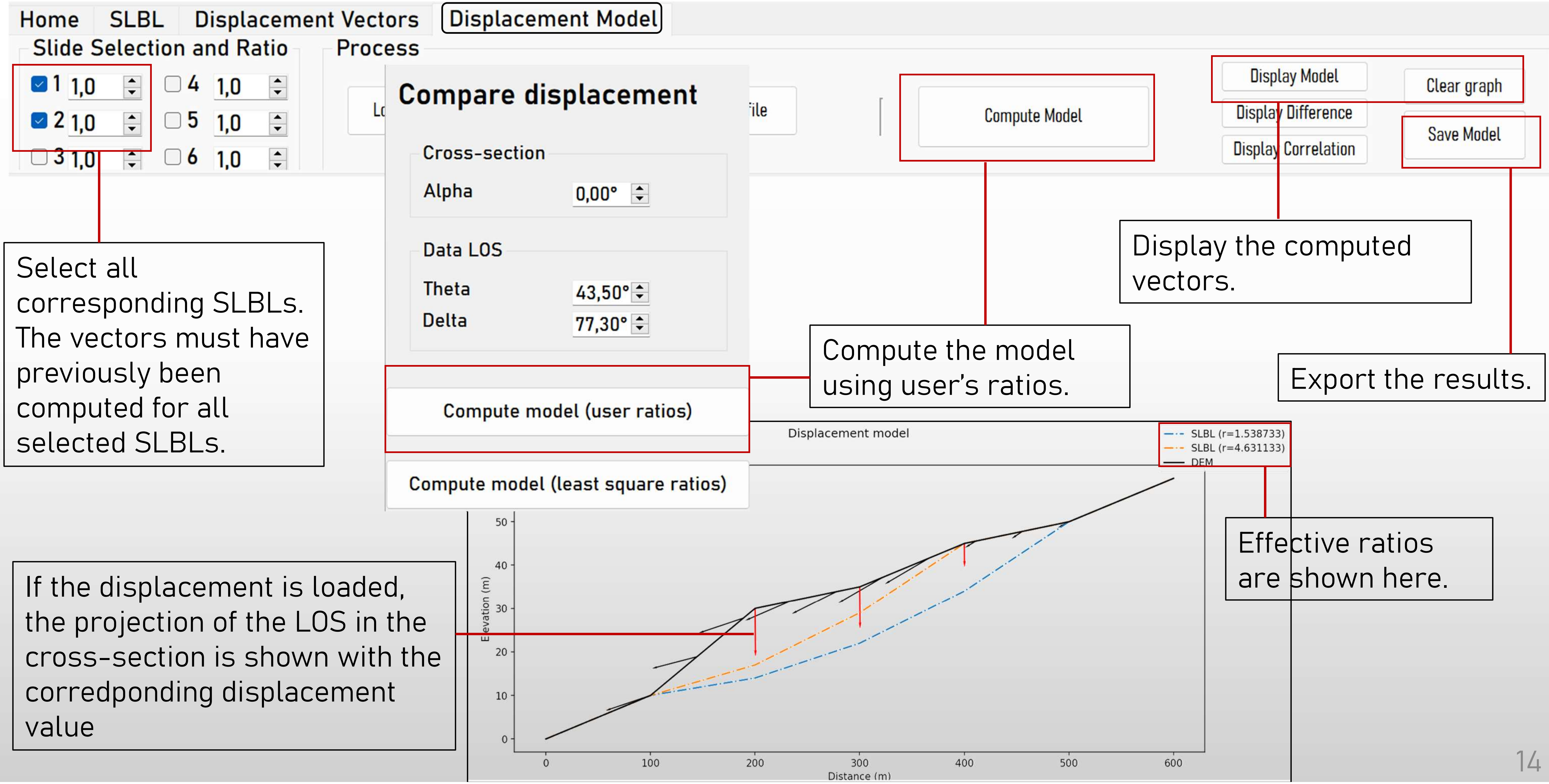
- With material deformation:
  - Incompressible but deformable material leads to a loss of thickness in the upper, whereas material is accumulating in the lower part (total volume is conserved)
    - Upper parts : decreasing thickness -> negative vertical displacement and positive horizontal displacement (relatively to the landslide)
    - Lower parts: increasing thickness -> positive vertical displacement and negative horizontal displacement



Modeling the displacement with one SLBL



Modeling the displacement with multiples SLBL



Select all corresponding SLBLs. The vectors must have previously been computed for all selected SLBLs.

Compute the model using user's ratios.

Display the computed vectors.

Export the results.

Effective ratios are shown here.

If the displacement is loaded, the projection of the LOS in the cross-section is shown with the corredponding displacement value

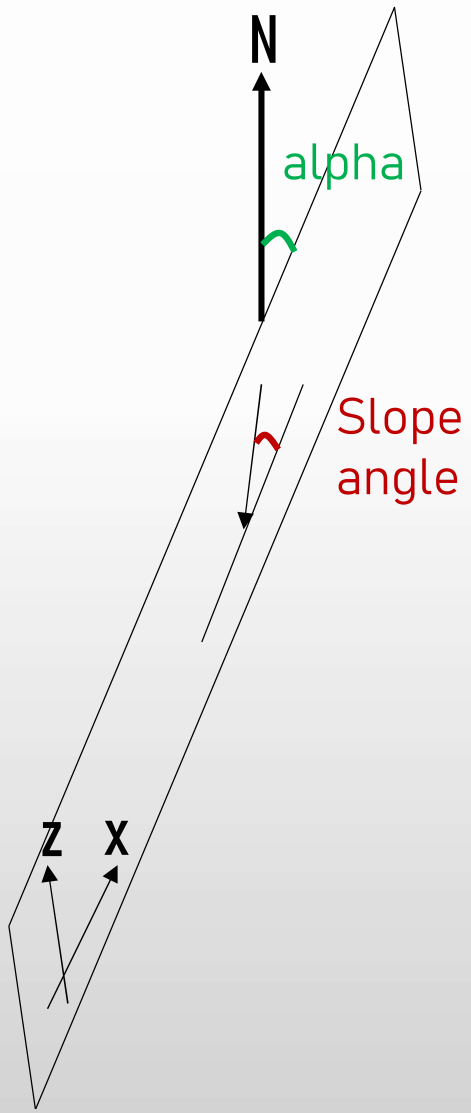
Recommandation

- If you want to change some parameters and do a new model, consider saving the current model into a project file, closing and opening again the python interface.

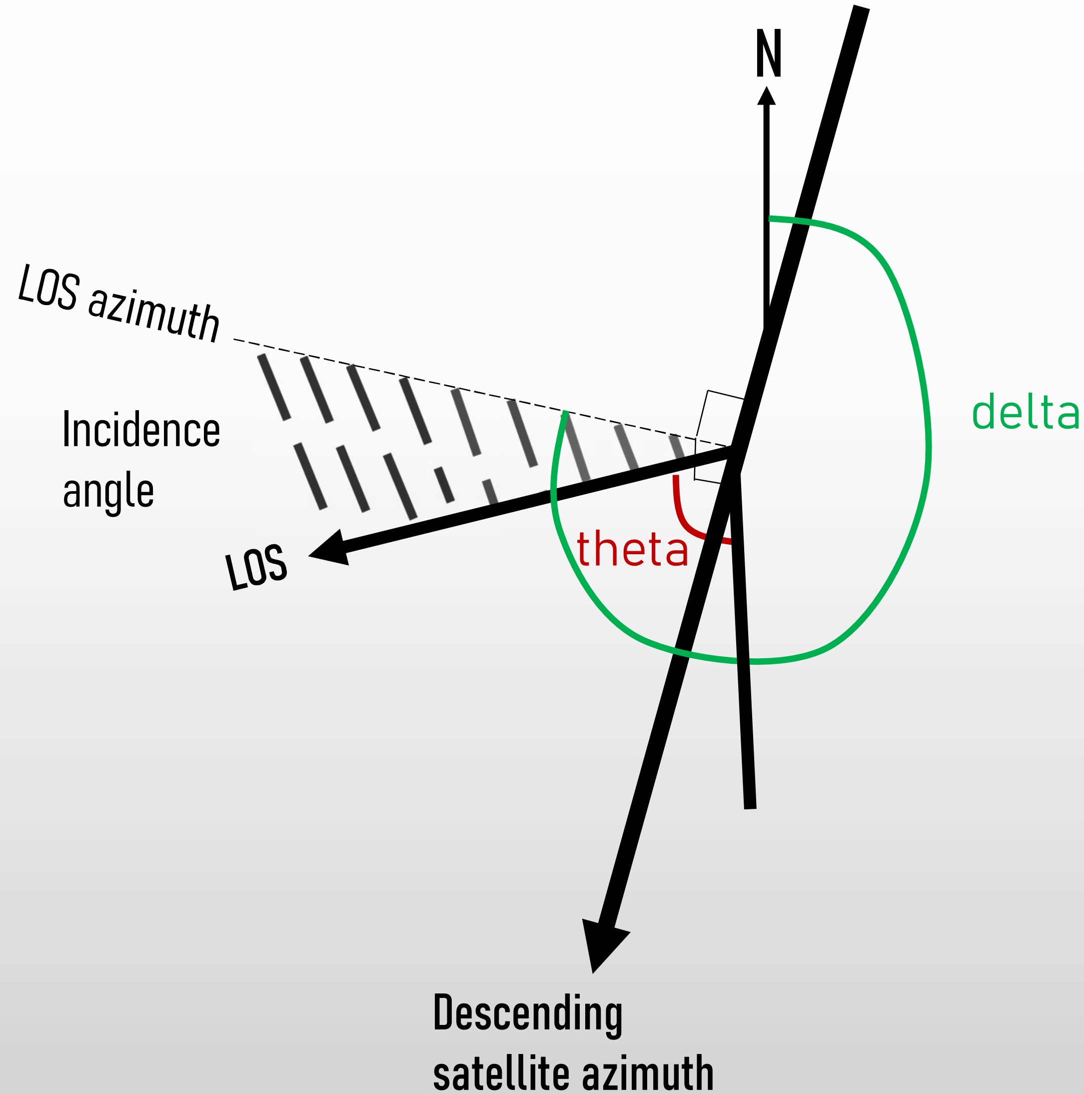
Reseting and reusing already used slots and parameters is currently not handled and could ends in unexpected behaviours or errors.

Comparing the displacement with field data : the geometry

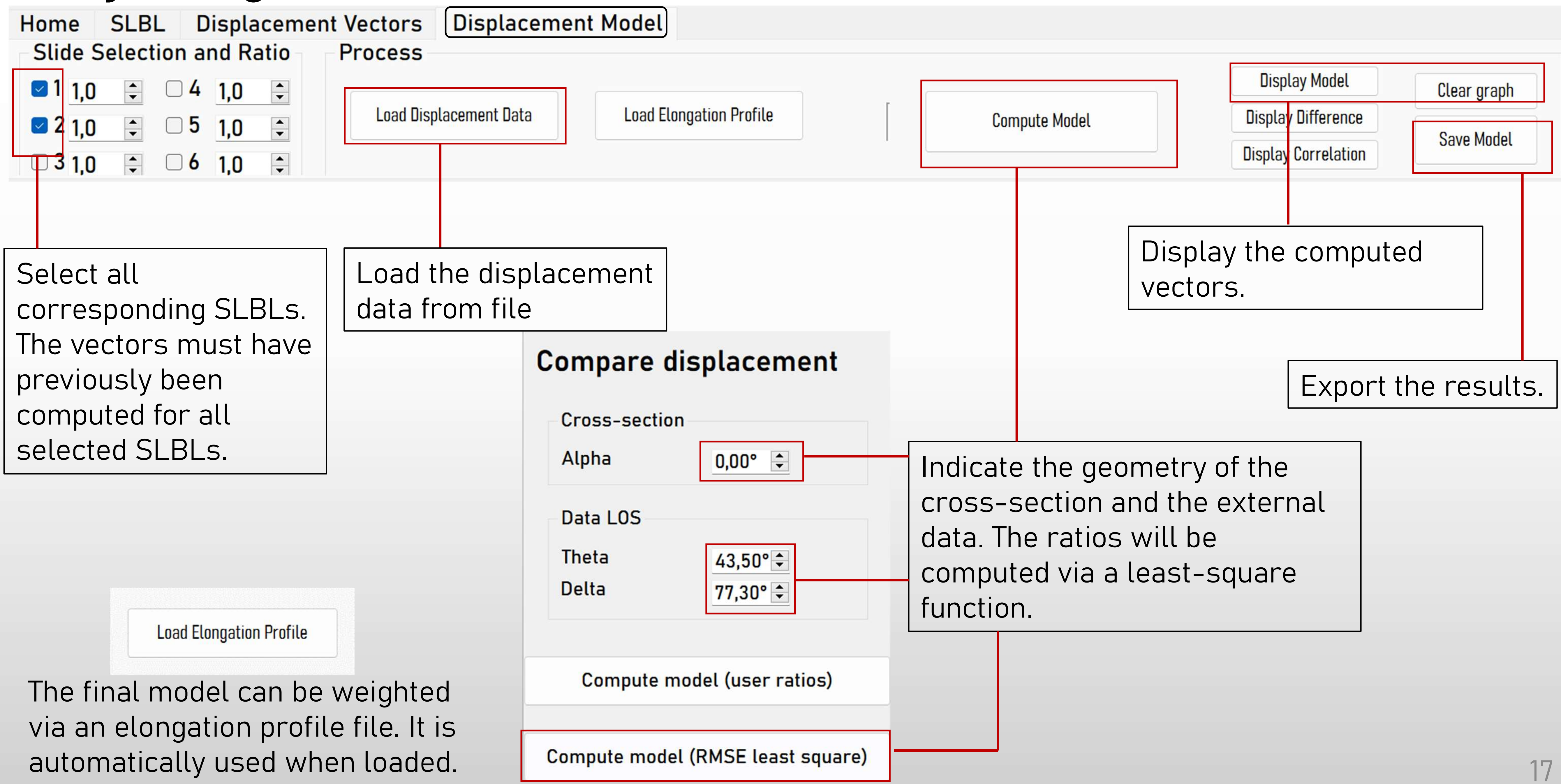
Cross-section



LOS



Ajusting the model to field data



Select all corresponding SLBLs. The vectors must have previously been computed for all selected SLBLs.

Load the displacement data from file

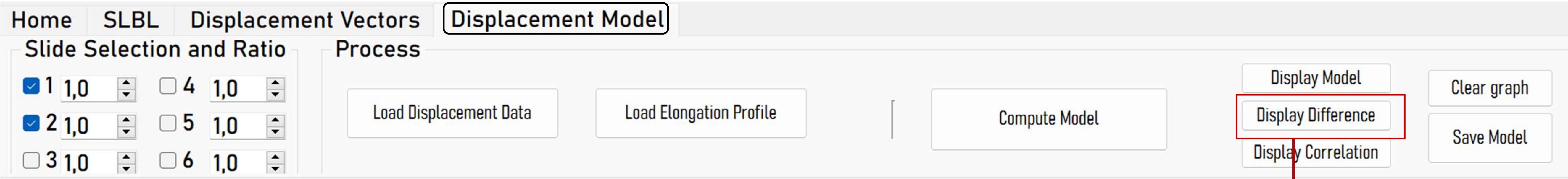
Display the computed vectors.

Export the results.

Indicate the geometry of the cross-section and the external data. The ratios will be computed via a least-square function.

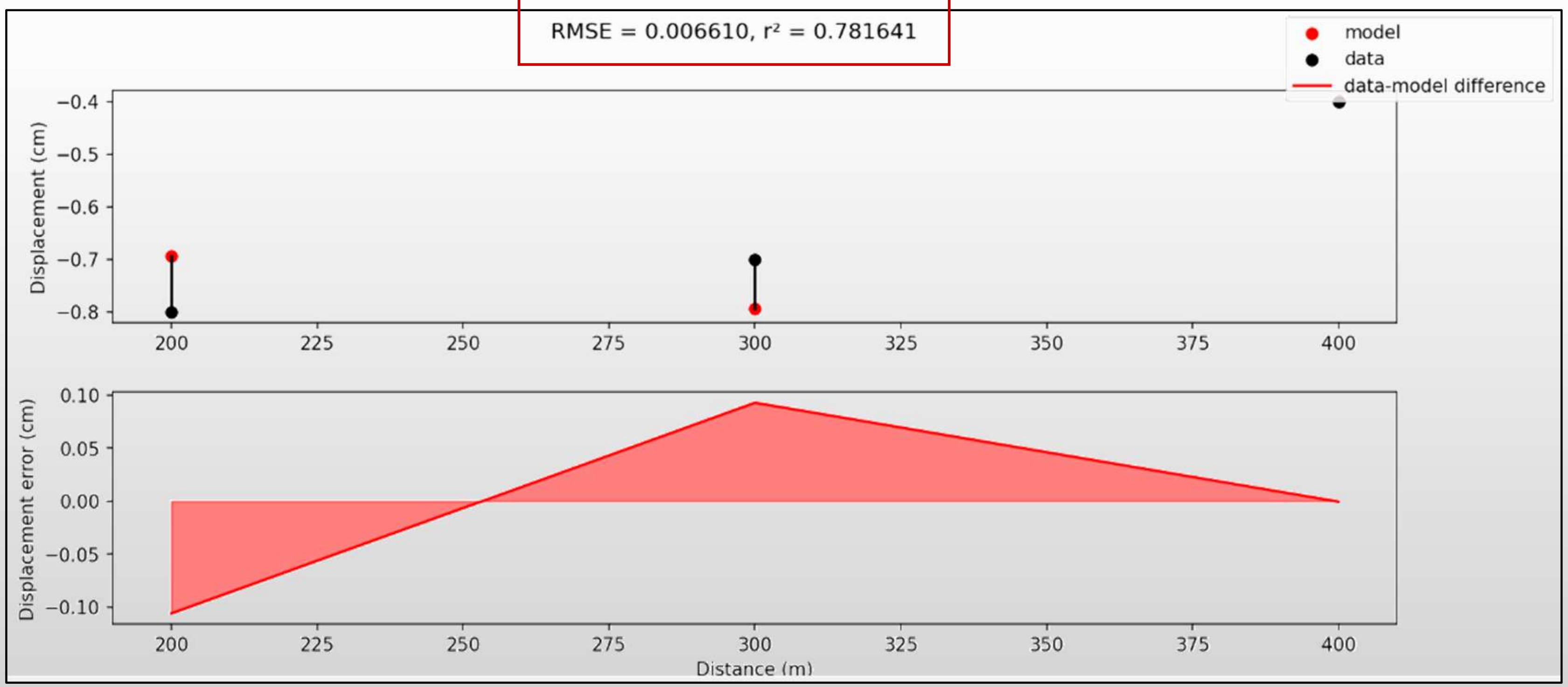
The final model can be weighted via an elongation profile file. It is automatically used when loaded.

Ajusting the model to field data



RMSE and r² coefficients are automatically computed and displayed.

After computing the model, the difference between the data and the model can be displayed.

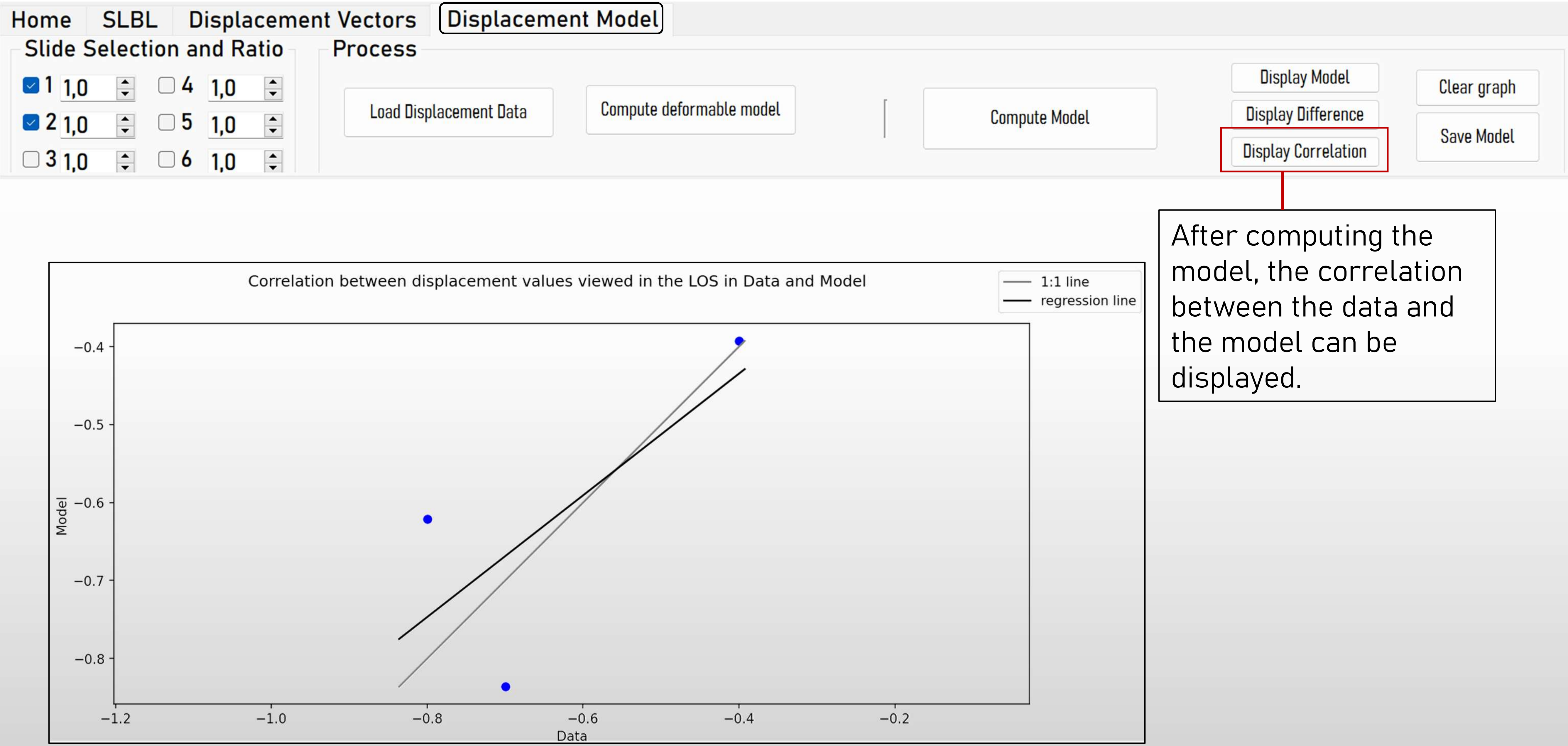


<u>Sign convention:</u>

-Positive displacement: Ground moving towards the satellite

-Negative displacement Ground moving away from the satellite

Ajusting the model to field data



After computing the model, the correlation between the data and the model can be displayed.

Model considering a deformable material at constant volume

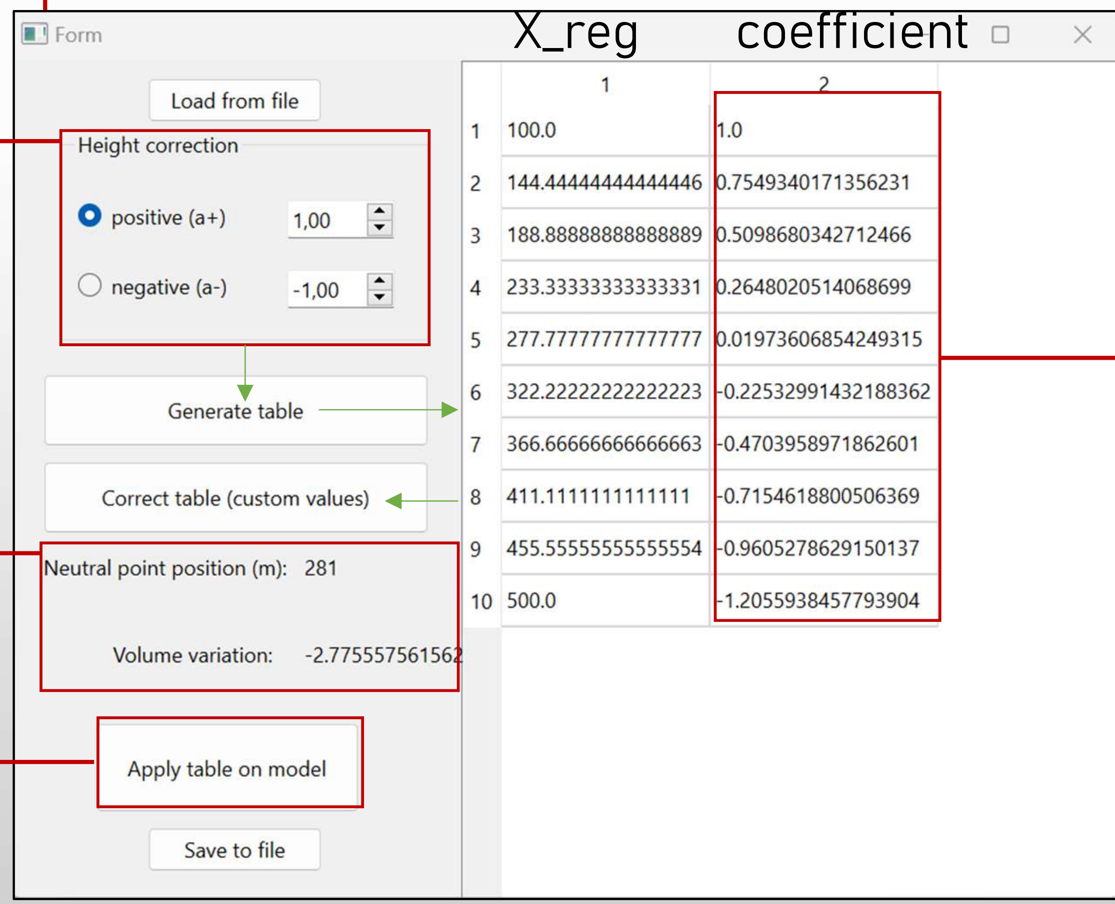
One of the two extremities

The neutral point is calculated considering the vertical displacement of the previous model :

$$\sum \vec{v}_{vertical} . coefficient = 0$$

Displacement is modified as:

$$\vec{v}_{vertical} \pm= \vec{v}_{vertical} . coefficient$$



	1	2
1	100.0	1.0
2	144.44444444444446	0.7549340171356231
3	188.88888888888889	0.5098680342712466
4	233.33333333333331	0.2648020514068699
5	277.77777777777777	0.019736068542493l5
6	322.22222222222223	-0.22532991432188362
7	366.66666666666663	-0.4703958971862601
8	411.1111111111111	-0.7154618800506369
9	455.55555555555554	-0.9605278629150137
10	500.0	-1.2055938457793904

Can be manually modified

Project file Exemple

```
GLOBAL
NAME test_project2
DEM_PATH path/example/dem.csv

SLIDE1
METHOD RANGE
LEFT_LIM 100
RIGHT_LIM 500
TOL 2.0

SLIDE2
METHOD RANGE
LEFT_LIM 100
RIGHT_LIM 400
TOL 2.0

DISP_VEC
N_VEC 10
X_MIN 100
X_MAX 500

DISP_MODEL
SLIDES 1;2
COEFFS 1.538732675989716;4.63113316590l916
METHOD LEAST_SQUARE
DISP_PATH path/example/disp.csv
ALPHA 12.0
THETA 35.0
DELTA 285.0

```