# Slope Stability Program (SSP)

*User's Guide 1.0*


**Contact:**

Brandon Karchewski
c/o Department of Civil Engineering - JHE-301
McMaster University
1280 Main St W
Hamilton, Ontario, Canada
L8S 4L7
*T:* 905-525-9140 x24287
*F:* 905-529-9688
*E:* karcheba@mcmaster.ca



8 January 2012

## Disclaimer

The user should note that the Slope Stability Program (SSP) is a research code under ongoing development. As such, the author makes no guarantee with respect to the accuracy of the results with regard to problems encountered in the field, as subsurface conditions are often uncertain at best. The program is provided to the user solely for educational purposes and is not intended for commercial use, engineering practice, or otherwise. Neither the author nor the associated institution are liable for decisions made based on the results of the program. If one requires further information, please contact the author directly (see contact details, email is the preferred form of communication).

# Table of Contents

# 1 PURPOSE

This guide introduces the use of the Slope Stability Program. The following sections provide a brief overview of the techniques employed in the program along with reference to appropriate sources from the literature. The final section presents a tutorial demonstrating the creation of an input file, operation of the program, and interpretation of the results. Note that this document is not an exhaustive reference manual; more details on the functions contained in the program can be obtained (as with built-in Matlab functions) by entering '`help <name of function>`' at the Matlab command line. If further information is required, please contact the author using the provided details (email is the preferred option).

# 2 INTRODUCTION

The Slope Stability Program (SSP) is a set of Matlab functions and scripts that analyzes slope problems using limit equilibrium analysis, which is a popular simplified technique used in geotechnical engineering to obtain the 'factor of safety' for the stability of a slope. Note that this form of engineering analysis, as with any model of physical reality, involves a number of assumptions that render the problem tractable. It is important to be aware of the assumptions made by the analysis technique, as engineering judgement is required in deciding whether the assumptions are valid for a given problem. The program contains one driver script (SlopeStabilityAnalysis.p) and three utility functions (GenAlgNonCirc.p, MorgPrice.p, RFEM.p), which are called in the same manner as Matlab m-files. The program is started by calling the driver script either by entering '`SlopeStabilityAnalysis`' at the command line or selecting 'Run' from the context menu for the file in the Matlab file browser. The following sections provide a brief description of what each file does, followed by an example problem demonstrating the use of the program from creation of the input file to interpretation of results.

# 3 MORGENSTERN-PRICE ANALYSIS

The file MorgPrice.p is a Matlab function that implements the Morgenstern-Price method of limit equilibrium slope stability analysis. The origination of the method dates to the mid-20th century (Morgenstern and Price, 1965) and is similar in principle to other classical limit equilibrium methods. It is considered to be one of the most versatile and robust methods in that it can handle slip surfaces of arbitrary shape and it satisfies both force and moment equilibrium. As in other limit equilibrium methods, the sliding mass is divided into a number of slices, assumptions are made about the forces acting on each slice, and a factor of safety is obtained by summing these forces appropriately. Figure 1 shows the free body diagram of a slice used in the implementation of the Morgenstern-Price method for the SSP.

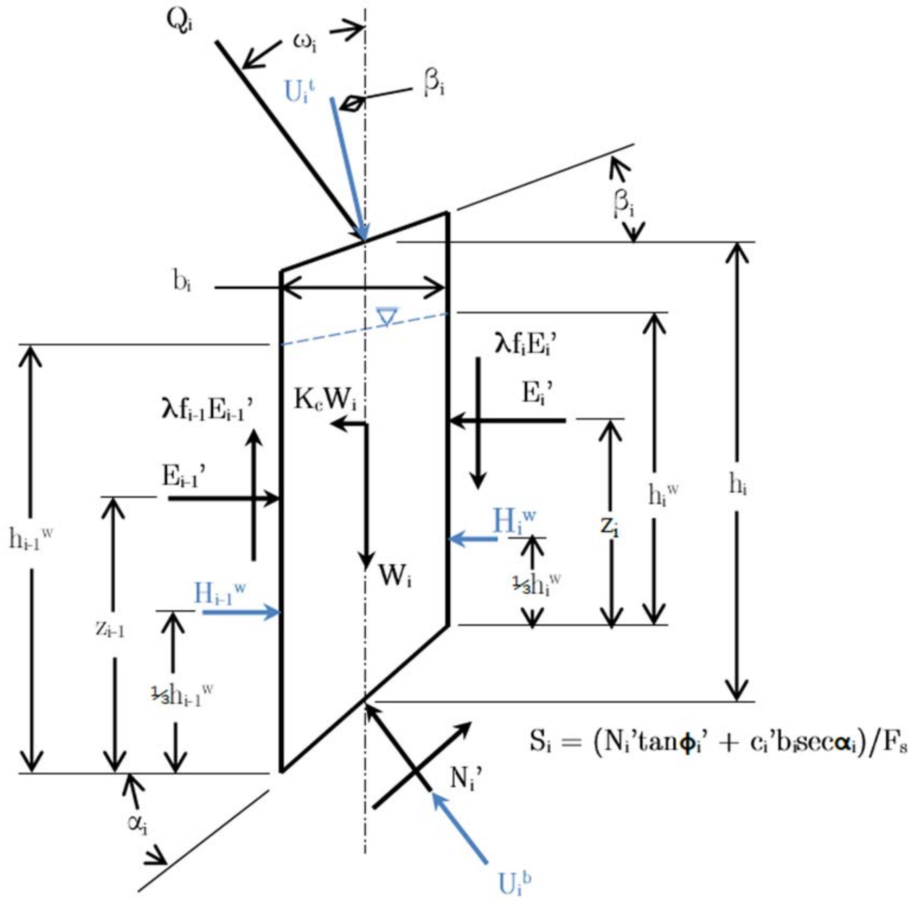


Figure 1. Free body diagram of a slice in the Morgenstern-Price method

At the base of each slice, there is an effective normal force, $N_i'$, and a shear force, $S_i$, which is related to the effective normal force through the Coulomb sliding law and a factor of safety, $F_s$. Note that this factor of safety is assumed to be the same for all slices and its value is determined through an iterative solution scheme. At each interface between slices there is an effective normal force, $E_i'$, and a shear force, $\lambda f_i E_i'$. The value of $f_i$ is obtained from an interslice force function, which in the SSP can be either a constant value of one or a half-sine function having a half-period equal to the width of the sliding mass and an amplitude of one. The value of $\lambda$ is determined through the iterative solution scheme (along with the value of $F_s$). The applied forces are those due to gravity (self-weight), $W_i$, water pressure at the base, $U_i^b$, water pressure at the surface, $U_i^t$, water pressure between slices (seepage force), $H_i^w$, surface loading, $Q_i$, and a quasi-static earthquake load, $K_cW_i$. Note that while all of these loads are implemented in the MorgPrice.p function, only self-weight and water pressure are accounted for in the driver program.

The algorithm used in MorgPrice.p is based on that presented by Zhu et al. (2005) with the main difference being that interslice water pressure forces are accounted for explicitly in MorgPrice.p.

The key assumptions in this analysis technique are the following:

i. The material properties within each soil layer are isotropic.
ii. Two-dimensional analysis is appropriate (*i.e.* the slope extends for a large distance in each relative to its height implying plane strain conditions).
iii. There is a direct relationship between the interslice normal and shear forces with the proportionality based on a constant factor, λ, and an interslice force function, f.
iv. There is a direct relationship between the normal and shear forces at the base of each slice given by the Coulomb sliding law and a constant factor of safety, $F_s$.

As the user of the SSP will not typically call this function directly, further details are not provided here. A more detailed description of the analysis method itself can be found in the provided references. Further information on the use of the MorgPrice.p function can be obtained by entering '`help MorgPrice`' at the Matlab command line. The shape of the input and output objects are indicated in the form [m,n] where m is the expected number of rows and n is the expected number of columns.

# 4 Rigid finite element method

The function RFEM.p implements the 'rigid finite element method' (RFEM) of limit equilibrium analysis as proposed by Stolle and Guo (2008). This technique also divides the sliding mass into slices, but the solution is obtained through a form of the finite element method where the slices are modelled as rigid bodies allowed to displace. One may regard the slices as 'nodes' in the analysis as the solution variables are the displacement of each slice in the horizontal and vertical directions. Interslice element stiffnesses are imposed in order to obtain the interslice normal and shear forces; note that these stiffnesses are non-linear and are related to the relative displacements of slices as normal stiffness is expected to be different in tension and compression for soils and the shear stiffness is expected to decrease as the material approaches failure. The boundary conditions are 'mixed' in the sense that they are applied as normal and shear stiffnesses at the base of each slice (i.e. at each node) with an appropriate transformation to the global coordinate system applied based on the angle of the base. The solution to the system is obtained through a modified form of the Newton-Raphson scheme combined with a bisection method for the load stepping (for cases where the global factor of safety is less than one). A local factor of safety for each slice is defined as the ratio between the available shear resistance (obtained through the Coulomb sliding law given the normal force at the base) and the actual (or mobilized) shear force at the base (computed from the displacement solution). This is an important advantage of RFEM, as it does not assume that the factor of safety is the same for every slice. The global factor of safety is the ratio between the sum of available shear resistances and the sum of mobilized shear forces at the base of all slices.

The key assumptions in this analysis technique, with the first two being the same as for the Morgenstern-Price method, are the following:

i. The material properties within each soil layer are isotropic.
ii. Two-dimensional analysis is appropriate (*i.e.* the slope extends for a large distance in each relative to its height implying plane strain conditions).
iii. The interslice and base stiffnesses are related to the displacement of the slices and have functional forms as proposed by Stolle and Guo (2008).

Again, since the normal use of the SSP does not require the user to call the RFEM function, further details are not provided here. A more detailed description of the RFEM can be found in the provided reference. More details on the implementation in the RFEM function can be obtained by entering '`help RFEM`' at the Matlab command line.

# 5 Genetic algorithm optimization

The methods of analysis described in the preceding sections compute the factor of safety for a particular surface given the geometry, stratigraphy, and material properties of the slope. However, there are an infinite number of possible slip surfaces that may be drawn through a given slope. The problem in slope stability analysis is to find the critical slip surface, that is, the slip surface that gives the lowest factor of safety. The function GenAlgNonCirc.p achieves this by employing a genetic algorithm search scheme similar to that proposed by Li et al. (2010). The genetic algorithm is a direct search scheme that uses a simplified model of evolution by natural selection to iteratively improve the guess at the location of the critical slip surface. In broad strokes, the algorithm works as follows:

1) The geometry, stratigraphy, and material properties of the slope are provided to the function along with an objective function to evaluate the factor of safety (the SSP uses the MorgPrice function) and feasible ranges for the geometry of the slip surface.

2) An initial 'population' of slip surfaces is generated based on the feasible ranges for the geometry of the slip surface and a set of reasonable heuristic principles built in to the GenAlgNonCirc function.
3) The factor of safety of each surface in the population is evaluated using the provided objective function.
4) The surfaces are sorted from lowest to highest factor of safety.
5) A set of 'parent' surfaces are selected based on a weighting scheme determined by the computed factors of safety.
6) 'Child' surfaces are generated from each pair of parent surfaces through a randomized interpolation process.
7) A small amount of random mutation is applied to the set of child surfaces.
8) The factor of safety is computed for each of the child surfaces.
9) The collection of parent and child surfaces is sorted from lowest to highest factor of safety.
10) The population for the next generation is filled through a randomized tournament selection scheme applied to the collection of parent and child surfaces.
11) Steps 5-10 are repeated until a set of stopping criteria are satisfied.

It is important to note that while the above process is very effective at seeking out the critical slip surface, the randomized nature of the search means that the same result is not obtained on every run. As such, one should normally conduct multiple runs to be sure that the minimum factor of safety has been found. Experience has shown that one can be relatively confident after 3 runs, but since a single run typically takes less than 20 seconds on a modern personal computer the author recommends that at least 5 runs be carried out for a given problem. Conducting multiple runs also has the advantage of allowing the user to observe how unique the critical failure surface is. That is, an analysis of the distribution of near-critical failure surfaces may also be performed.

Further details on the form of genetic algorithm used by the SSP may be obtained in the provided reference. A more detailed description of the GenAlgNonCirc.p function may be obtained by entering '`help GenAlgNonCirc`' at the Matlab command line.

# 6 Driver program

The SlopeStabilityAnalysis.p script is the driver program for the SSP. The SSP is typically run by calling this script in the Matlab workspace and providing the requested details. The script proceeds as follows:

1) Collect input data from the user.
2) Read geometry, stratigraphy, and material properties from the input data file.
3) Run a genetic algorithm search for the critical slip surface using MorgPrice as the objective function.
4) Reanalyze the critical surface using the RFEM.
5) Plot the critical surface, displaced slices (from the RFEM solution), global and local factors of safety, and interslice forces (from both MorgPrice and RFEM).
6) Generate an output file containing the same information provided by the plots in step 5. If an output file from a previous analysis of the same problem already exists, the output data are amended to the end of this file.

As mentioned above, the author recommends at least 5 runs of the SSP for a given problem to be reasonable certain of the minimum factor of safety and the location of the critical slip surface. A tutorial demonstrating the workflow for a typical slope stability problem follows.

# 7 Tutorial problem

The preceding sections have provided a general overview of how the SSP solves a slope stability problem. This section provides a practical example of how to set up such a problem and solve it using the program. Figure 2 shows the geometry of a slope stability problem with two soil layers and a water table as shown by the dashed line. Such a problem might represent the foundation of linear infrastructure (*e.g.* highways, railroads, etc.) or an earth embankment for a dam.

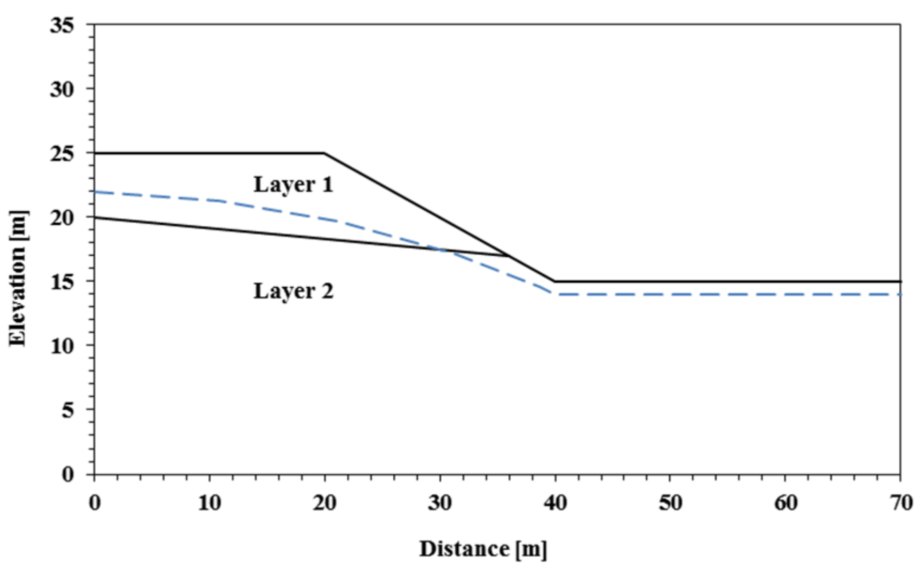


Figure 2. Tutorial slope stability problem

Table 1 shows the material properties of each soil layer where φ' is the effective angle of friction, c' is the effective cohesion, γ is the unit weight, $\gamma_{sat}$ is the saturated unit weight, E is the elastic modulus, and ν is Poisson's ratio. The unit weight of water is assumed to be 9.8 kN/m³.

**Table 1.** Material properties for tutorial problem

| **Layer** | **φ'** [deg] | **c'** [kPa] | **γ** [kN/m³] | $\boldsymbol{\gamma_{sat}}$ [kN/m³] | **E** [kPa] | **ν** |
|---|---|---|---|---|---|---|
| 1 | 20 | 5 | 15 | 15 | 15000 | 0.4 |
| 2 | 25 | 10 | 18 | 18 | 12000 | 0.37 |

Table 2 shows the geometric coordinates of the soil layers and the water table. Note that for all sets of coordinates, the x-coordinates proceed monotonically from left to right and the minimum and maximum x-coordinates are the same. Also, note that where Layer 1 is not present, its coordinates are the same as Layer 2. In this way, Layer 1 is "pinched out" to zero thickness rather than not represented at all in this portion of the domain. All coordinate sets provided to the SSP must meet these criteria for proper analysis of the geometry.

**Table 2.** Geometry of tutorial problem

| **Layer** | **x** [m] | **y** [m] |
|---|---|---|
| 1 | 0.00 | 25.00 |
| | 20.00 | 25.00 |
| | 30.00 | 20.00 |
| | 40.00 | 15.00 |
| | 70.00 | 15.00 |
| 2 | 0.00 | 20.00 |
| | 36.00 | 17.00 |
| | 40.00 | 15.00 |
| | 70.00 | 15.00 |
| water | 0.00 | 22.00 |
| table | 10.87 | 21.28 |
| | 21.14 | 19.68 |
| | 31.21 | 17.17 |
| | 38.69 | 14.56 |
| | 40.00 | 14.00 |
| | 70.00 | 14.00 |

The input data file for a given slope stability problem is built as follows:

1) The first line indicates the number of soil layers and the direction of soil movement separated by one or more whitespace characters (either space or tab). The direction of soil movement is entered as 1 if the movement is left-to-right (as in this example) and 0 if the movement is right-to-left. For the tutorial problem, this line should read:

```
2       1
```

2) The next line indicates the number of geometry points in the first soil layer followed by its material properties in the same order as given in Table 1. For example, for Layer 1 this line should read:

```
5    20       5     15    15  15000   0.4
```

3) The following series of lines provide the x and y coordinates of the soil layer with one line per coordinate pair. For example, for Layer 1 this would read:

```
0       25
20      25
30      20
40      15
70      15
```

4) Repeat steps 2 and 3 for each soil layer.
5) Once all soil layers have been entered, the next line indicates the number of geometry points for the water table. If the slope is dry (*i.e.* there is no water table), this line would just have a 0 and it would be the last line in the file. For the tutorial problem, this line should read:

```
7
```

6) If there is a water table, the next line gives the unit weight of water. For the tutorial problem, this line should read:

```
9.8
```

7) The following lines give the x and y coordinates of the water table in the same format as the geometry of the soil layers.

The file 'Ex3.dat' provided with the program contains the complete input data file for the tutorial problem. The reader is encouraged to have a look at it before moving on to make sure that the structure of the file is clear. Note that there should be no extra blank lines or additional text in the file besides the values as described above. When setting up new data files, it is useful to enter the data in a spreadsheet program with appropriate labels for the data and then copy the values into a text file when the data set is complete. The file extension can be anything except '.out' or '.OUT' as these are reserved for the output file that is generated by the SSP.

Once the input file has been created, place a copy into the program directory, and start Matlab (setting the working directory to the directory containing the SSP and the data file). Begin the analysis by entering:

```
≫ SlopeStabilityAnalysis
```

at the Matlab command line. The program will first ask for the name of the data file:

```
Enter the name of the slope data file (including extension):
```

Enter the name of the input data file. For the tutorial problem, this should read:

```
Ex3.dat
```

The program will now ask for reasonable ranges for the x-coordinates of the entry and exit points for the slip surface. The entry range should always be to the left of the exit range. In slope stability problems, one is typically concerned with relatively deep-seated failure surfaces passing through regions near the crest and toe of the slope since shallow failures are often prevented by surface vegetation. For example, in this tutorial problem reasonable estimates for the domain of entry and exit coordinates are [10 m, 24 m] and [34 m, 52 m], respectively. In the Matlab workspace, this would look like:

```
Enter minimum x-coord for slip surface entry: 10
Enter maximum x-coord for slip surface entry: 24

Enter minimum x-coord for slip surface exit: 34
Enter maximum x-coord for slip surface exit: 52
```

Next, the program will ask for the feasible range for the y-coordinates of the slip surface. This should typically range from a line well below the toe of the slope (perhaps below any layer of relatively weak material, if present) to a line at or above the crest of the slope. For the tutorial problem, a reasonable range is [5 m, 26 m]. In the Matlab workspace, this would look like:

```
Enter minimum y-coord for slip surface: 5
Enter maximum y-coord for slip surface: 26
```

Finally, the program will ask for the interslice shear force distribution. Enter 1 to use a constant value of 1 or 0 to use a half-sine function as described in Section 3. For the tutorial problem, selecting the half-sine option would look like:

```
Select interslice shear force distribution,
1 => f(x) = 1 (Spencer's method)
0 => f(x) = half-sine (standard Morgenstern-Price): 0
```

The program will now run and seek the critical slip surface. Upon completion, a set of plots as shown in Figure 3 will be generated showing the location of the critical failure surface for this run, the displaced slices from the RFEM analysis (magnified by the given factor), the factors of safety obtained by the two analysis methods, and the distribution of interslice forces for comparison. An output file is also generated with the same name as the input file, but with the extension '.out' containing the same information in tabular form.

The reader should note that when running the program for the same problem, the results will be similar, but not necessarily the same. Slight variations in the factor of safety obtained on each run are to be expected and, in fact, provide insight into the uniqueness of the critical slip surface. Also, note the distribution of local factors of safety obtained from the RFEM analysis. In this problem, the lowest factors of safety occur near the crest of the slope where the largest displacements occur. In other problems, one may also note separation of slices near the crest indicating the development of a 'tension crack'. The distribution of interslice normal forces shows good agreement between the Morgenstern-Price and RFEM analyses, but the interslice shear force distribution is quite different. Interslice shear forces only occur where there is relative displacement in the vertical direction between slices. Since the mass is tending to slide in a rotational manner, large values for interslice shear forces only occur where there is a significant change in material properties such as where the slip surface intersects the water table or a transition between soil layers. This lack of mobilized shear between slices means that the Morgenstern-Price method seems to have over predicted the factor of safety in this case since it does not capture this phenomenon. The more conservative estimate for the factor of safety is, in this case, that provided by the RFEM analysis.

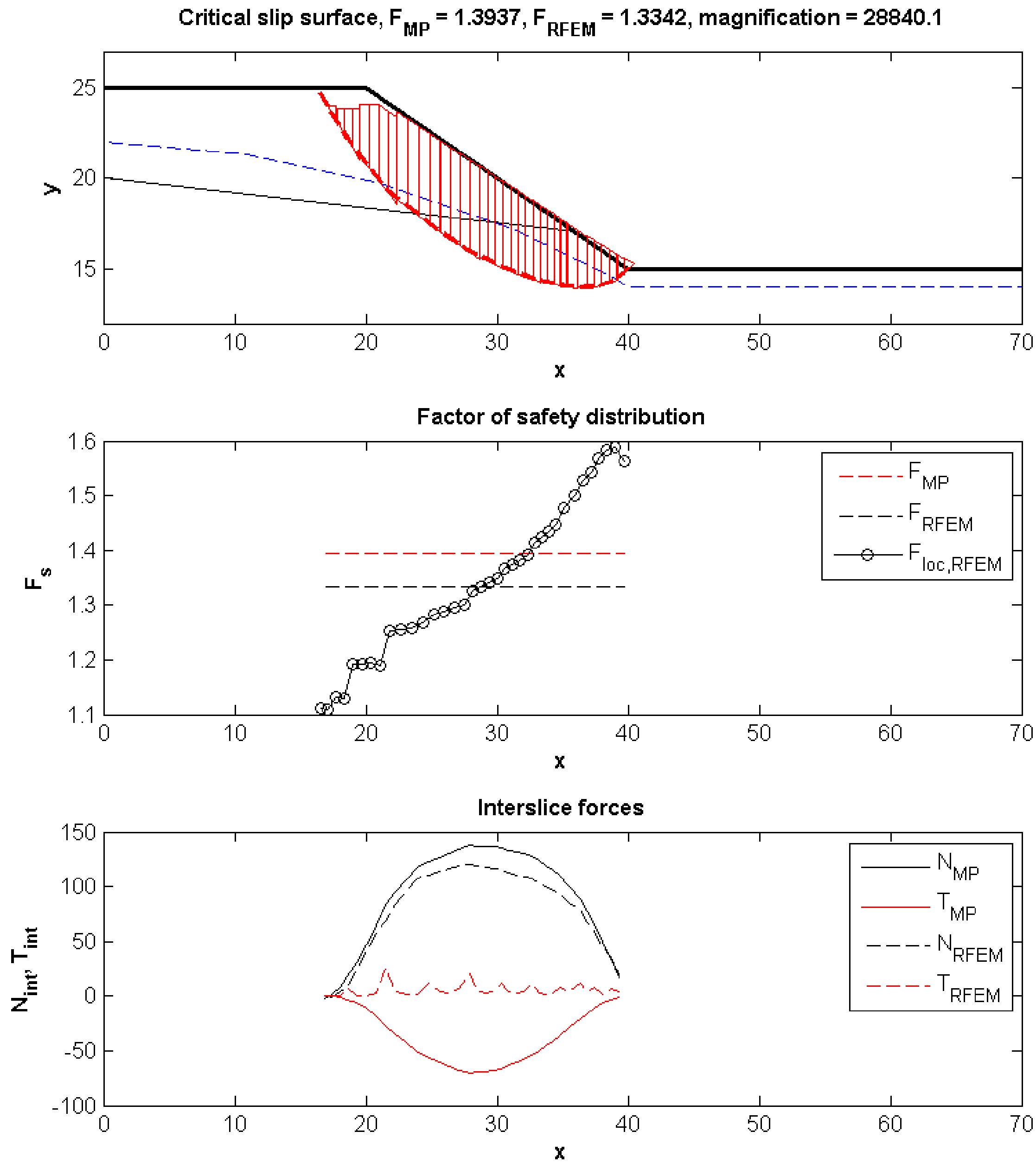


**Figure 3.** Sample output for tutorial problem

# 8 Conclusion

This concludes the overview of the use of the Slope Stability Program. Proficiency with the program is best obtained through experimentation, so the reader is encouraged to try solving similar problems on their own. As mentioned in the Disclaimer, this is a research code under ongoing development, so it is possible that there may be cases that generate errors or spurious output. Any feedback (positive or negative) is welcomed by the author. Enjoy!

# 9 References

Li,Y.C., Chen,Y.M., Zhan,T.L.T., Ling,D.S., and Cleall,P.J. (2010). An efficient approach for locating the critical slip surface in slope stability analyses using a real-coded genetic algorithm. *Canadian Geotechnical Journal*, 47(7), 806-820.

Morgenstern, N.R. and Price, V.E. (1965). The analysis of the stability of general slip surfaces. *Geotechnique*, 15(1), 79-93.

Stolle, D. and Guo, P. (2008). Limit equilibrium slope stability analysis using rigid finite elements. *Canadian Geotechnical Journal*, 45(5), 653-662.

Zhu,D.Y., Lee,C.F., Qian,Q.H., and Chen,G.R. (2005). A concise algorithm for computing the factor of safety using the Morgenstern-Price method. *Canadian Geotechnical Journal*, 42(1), 272-278.