1 Slope Stability for a Cohesive and Frictional Soil

1.1 Problem Statement

A common problem encountered in engineering soil mechanics is the stability of soil slopes in frictional materials. In this example, three slope conditions are analyzed. First, a slope in sand with zero cohesion is modeled with an initial slope steeper than the angle of repose of the sand. This slope, of course, should collapse; the progression of this collapse is calculated as it develops. Second, a small cohesion is added to the material and the slope is reexamined to determine whether it is stable. Third, the water level in the slope is raised, and the effect on stability is examined.

In this example, the soil is homogeneous, and the stability and factor of safety of the slope can readily be determined using an analytic or graphical technique.* However, the power of the *FLAC* code is in its ability to model more complex slope geometries in which, for example, several layers of soil with differing material properties and/or constitutive behaviors may exist. This type of problem can be examined with no greater effort than needed for the homogeneous case by simply assigning different material properties and/or models to different zones.

This example also demonstrates two approaches to analyze the effect of a phreatic surface in the slope. In one approach, an effective-stress analysis is performed after adding a pore-pressure distribution directly to the zones in the model. In the other approach, a groundwater flow calculation is performed first to establish the phreatic surface; then the effective-stress analysis is performed.

1.2 Modeling Procedure

1.2.1 Initial Model State

The following sequence of operations is used to initialize the slope and create a restart file from which the boundary conditions and/or material properties may be varied for the three cases of this example. Refer to the project file “slope.prj” for a complete description of this example.

An initial grid of 20×10 square zones is set up. Note that this initial grid is assigned the dimensions of 20 units in the x-direction by 10 units in the y-direction, unless otherwise redefined by the user. The command to do this is

```
grid 20 10
```

* See the **Factor of Safety** volume for comparisons of *FLAC* results to limit equilibrium analysis.

The basic grid is deformed into the shape of a slope and the soil base beneath it. This is done using the **GENERATE** command. Three different quadrilateral regions are created with three **GENERATE** commands:

```
gen 0,0  0,3  5,3  5,0    i = 1,6  j = 1,4
gen 5,0  5,3  30,3  20,0  i = 6,4  j = 1,4
gen same 9,10 20,10 same  i = 6,21  j = 4,11
```

The first two **GENERATE** commands define the base of the slope, and the third **GENERATE** command creates the slope. Note that the zones are aligned with the angle of the slope so that the zones along the slope face are all quadrilateral-shaped. This is recommended because all zones are then composed of two overlaid sets of triangular elements. These zones are well-suited for plasticity analysis (see Section 1.3.3.2 in **Theory and Background**). It is also possible to create a slope using the **GENERATE line** command. However, with this command, single triangular zones will be created along the slope face; these zones are not as accurate for plasticity analysis.

The **GROUP** command is used to assign the group name **soil** to all the zones.

A Mohr-Coulomb constitutive model is assigned to the **soil** zones with the following properties.

density	1500 kg/m^3
shear modulus	0.3×10^8 Pa
bulk modulus	10^8 Pa
friction angle	20°
cohesion	10^{10} Pa
tensile strength	10^{10} Pa

Note that a high cohesion and tensile strength are assigned to prevent slope failure during the initialization of gravitational stresses in the model (see above).

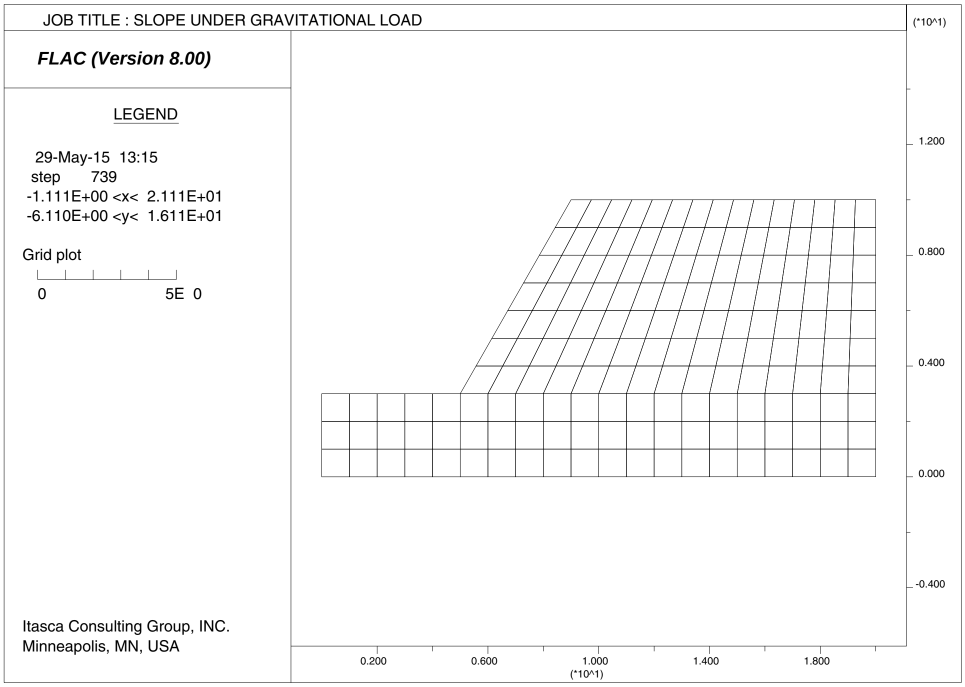

Figure 1.1 Grid plot of initial slope

The boundary conditions consist of roller boundaries on the left and right sides of the model, as well as a fixed base. The acceleration of gravity is set to 9.81 m/sec^2 (positive means acting downward). An initial elastic state in which gravitational stresses are equilibrated is desired. This is achieved with the **SOLVE** command, using default limits. Equilibrium is obtained when the out-of-balance force ratio limit of 10^{-3} is reached. To examine the progression of the solution, the y-displacement history is requested at a gridpoint at the slope crest. This is done using the **HISTORY** command. When the **SOLVE** command has reached its limit, the history may be plotted to verify that the mesh is indeed at an equilibrium state (i.e., **PLOT history 1**). A numerical and graphical representation of the elastic stresses can be obtained by issuing the commands

```
print sxx syy
plot sxx  yellow  bound
plot syy  yellow  bound
```

Initially, very large values for the cohesion and tensile strength are assigned to the slope. In justification, let us reexamine the way an explicit model works. An initial grid is created first and, in this case, gravity applied to the gridpoints and gravitational stresses allowed to equilibrate. For most problems, it is desirable that this process occur as rapidly as possible. This can be done by requiring the material to behave elastically during the equilibration process.

Once stresses have equilibrated, actual material properties are assigned, excavation is made, loads are applied, etc., and the simulation process continued. In the case illustrated here, a plastic constitutive model is assigned initially, with high cohesion and tensile strength, forcing the material to behave elastically. Then the cohesion and tensile strength are reset to the desired values. This eliminates the necessity of redefining the constitutive model and properties after equilibrium, thus saving a few steps of input.* The same effect could be obtained by using an elastic model initially for the equilibration process, followed by a change to the Mohr-Coulomb model, followed by a definition of the material friction, cohesion and tensile strength. Either methodology for initializing a gravity-loaded grid is acceptable, and is selected based on the preference of the user.

A restart file is created to save the elastic equilibrium state. This is done to save time in case future runs will be made in which material parameters or constitutive models are varied. Performing these studies requires only that the elastic state be restored, therefore eliminating the necessity of recomputing the equilibrium state.

The command

```
save sl1.sav
```

will create a restart file on the default drive called "SL1.SAV." *FLAC* could be halted at this point, and the program run with the saved state restored at a later time, simply by typing

```
restore sl1.sav
```

1.2.2 Slope Collapse: Dry Conditions

For the next stage of the simulation, the material properties are set to the actual soil values and the calculation continued while examining the possible failure process. During this process, plots of the progressive displacement of the slope are made. To avoid any confusion in analyzing the data, only the *change* in displacement is monitored, not the *cumulative* displacements from the beginning of the simulation. The calculation procedure in *FLAC* does not involve displacements, but keeps the cumulative total for each gridpoint as a convenience to the user. Therefore, the displacements may be initialized at any point in the run without affecting the results. This is done by using the command

```
ini  xdis=0  ydis=0
```

From this point on, plots or printouts will show only the *change* in displacement from the previous state.

Next, the material properties of the zones are reset by using the **PROPERTY** command. The cohesion is set to zero for all zones that are currently composed of soil.† Finally, the calculation mode is set to *large-strain* to provide a more accurate geometrical representation of the slope failure as it progresses. Because slope collapse will occur due to the angle of repose of the soil being smaller than the slope angle, the **SOLVE** command is not used (because equilibrium will not be reached). The **STEP** command is used to step the simulation a small number of calculational steps at a time,

---

* The **SOLVE elastic** command performs this two-step operation automatically.

† Note that the tensile strength will also be set to zero in *FLAC*, because the tensile strength is calculated from cohesion/tan (friction).

stopping to print and plot intermediate stages. Here, the power of the explicit method is evident in its ability to follow highly nonlinear problems, which may never converge to an equilibrium state, through progressive failure.

The model is now stepped in intervals to evaluate the progressive collapse of the slope. The collapse is revealed when printing and plotting results after each step. The following commands are used.

```
print xv yv xd yd state
plot xv z disp bou
```

Figures 1.2 and 1.3 show the state of the slope after 200 steps and after an additional 800 steps. These figures illustrate the progressive collapse and, in particular, indicate the location of the failure (slip) surface. The slope is collapsing in an attempt to reach its angle of repose. At some point, the displacements of the gridpoints become unrealistic because of extreme distortion of the grid. *FLAC* automatically checks for excessive grid deformation and will stop the calculation process if the condition is detected, displaying an error message.

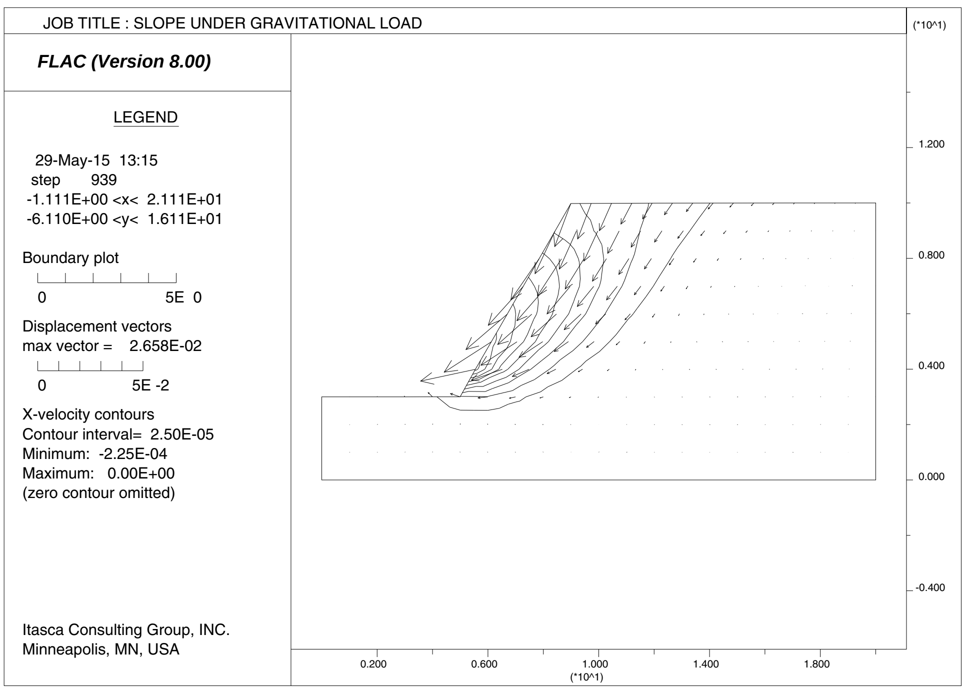

Figure 1.2 Plot of displacement vectors and x-velocity contours after 200 steps with zero cohesion

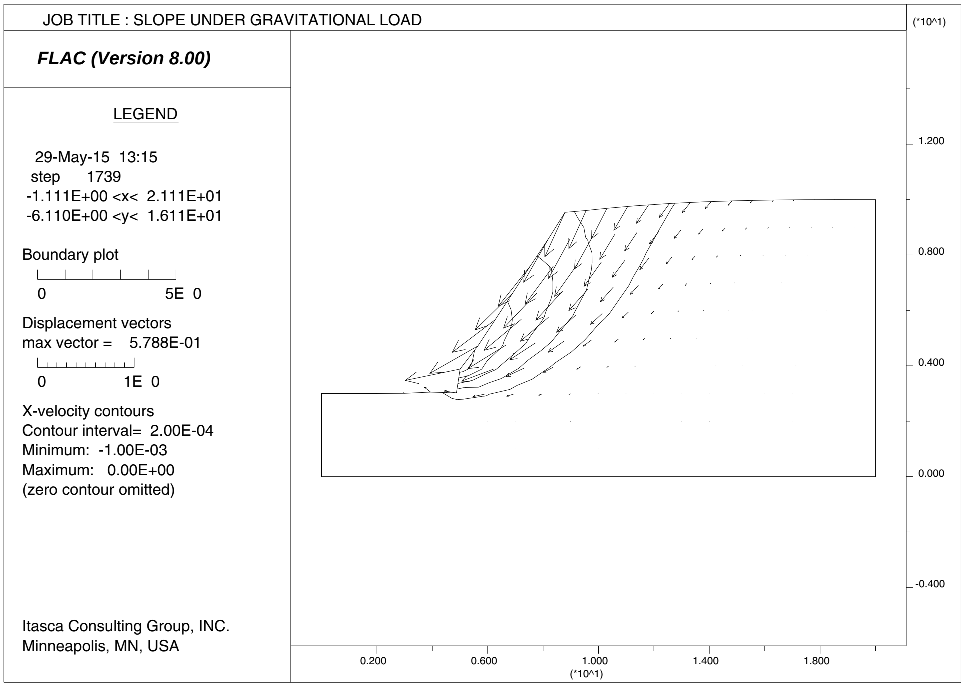

Figure 1.3 Plot of displacement vectors and x-velocity contours after 1000 steps at zero cohesion

An interesting comparison can be obtained by running another simulation in which a small cohesion and zero tensile strength are assigned to the soil. Because the initial elastic equilibrium state has been saved, the problem can be restored from this state:

```
rest sl1.sav
```

Following the procedure used earlier, enter the commands

```
ini  xdis=0  ydis=0
prop  coh=1e4  tens=0.0
set  large
solve
```

Under these conditions, the results will show that the slope is stable. (Note the small magnitude of the calculated displacements.)

1.2.3 Effective Stress Analysis with **WATER table**

Next, we wish to assess the stability of the slope with a water table present. Continuing with the model at the present state, the water level is raised in the slope to a height of 9 m on the right side of the model and 5 m on the left side (i.e., 2 m above the base of the excavation). It is a simple matter to use the **WATER table** command to specify the phreatic surface. Note that the correct wet and dry densities must also be supplied. The *FISH* function **wet_den** prescribes the wet density to zones below the water table, assuming a soil porosity of 0.3. The **APPLY pressure** command must also be used to apply the weight of the water in the excavation to the surface of the excavation. Figure 1.4 illustrates the location of the water table, the applied forces representing the weight of the water in the excavation, and the wet and dry densities in the zones.

The calculation is continued with the **STEP** command. Now, after an additional 6000 steps, the slope is observed to fail. The pore pressure distribution and velocity vectors indicating the slope failure in the model are plotted in Figure 1.5.

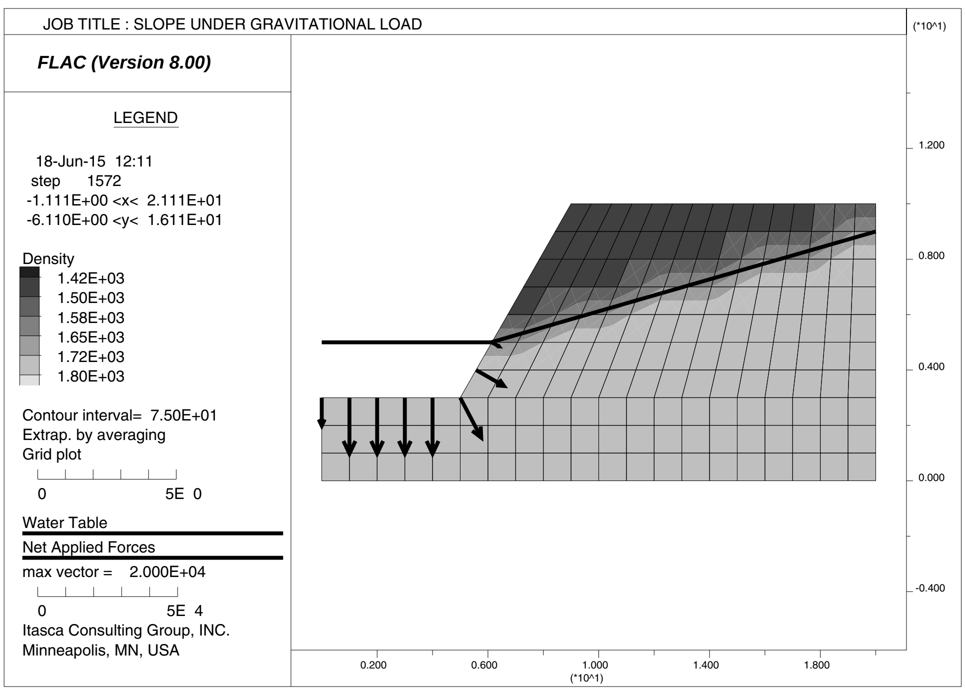

Figure 1.4 Location of water table, applied forces along slope and wet and dry densities

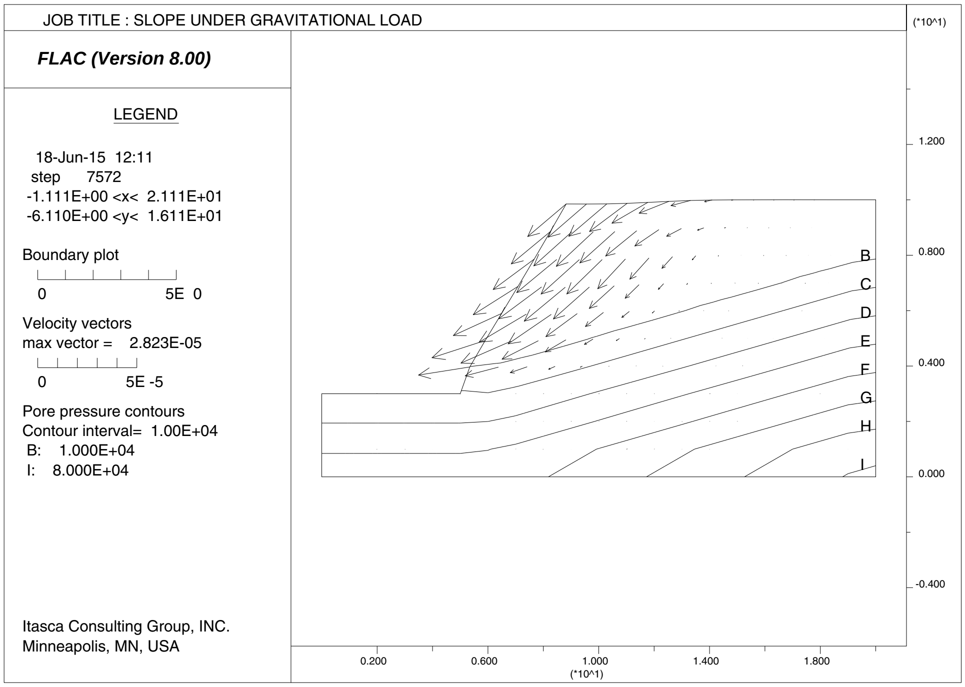

Figure 1.5 ***Pore pressure distribution and velocity vectors in failed slope (using* WATER table*)***

1.2.4 Effective Stress Analysis with **CONFIG gwflow**

Alternatively, the groundwater flow option in *FLAC* can be used to find the phreatic surface and establish the pore pressure distribution before the mechanical response is investigated. The model is run in groundwater flow mode by using the **CONFIG gwflow** command. This command must be given before the **GRID** command. We turn off the mechanical calculation (**SET mech off**) in order to establish the initial pore pressure distribution. We apply pore pressure boundary conditions to raise the water level to 5 m at the left boundary and 9 m at the right. The slope is initially dry (**INITIAL saturation 0**). We also set the bulk modulus of the water to a low value (1.0×10^4) because our objective is to reach the steady-flow state as quickly as possible. The groundwater time scale is wrong in this case, but we are not interested in the transient time response. The steady-flow state is determined by using the **SOLVE** command. When the groundwater flow ratio falls below the set value of 0.01, steady-state flow is achieved. This can be checked by using the *FISH* function “QRATIO.FIS” to assess the flow state. (See Section 3 in the ***FISH* volume** for a description of this function.) The steady-flow state is indicated by the plot of flow vectors and phreatic-surface contour in Figure 1.6. (**PLOT water** is used to plot the phreatic surface contour.) The bumpy phreatic-surface line is due to the coarse discretization.

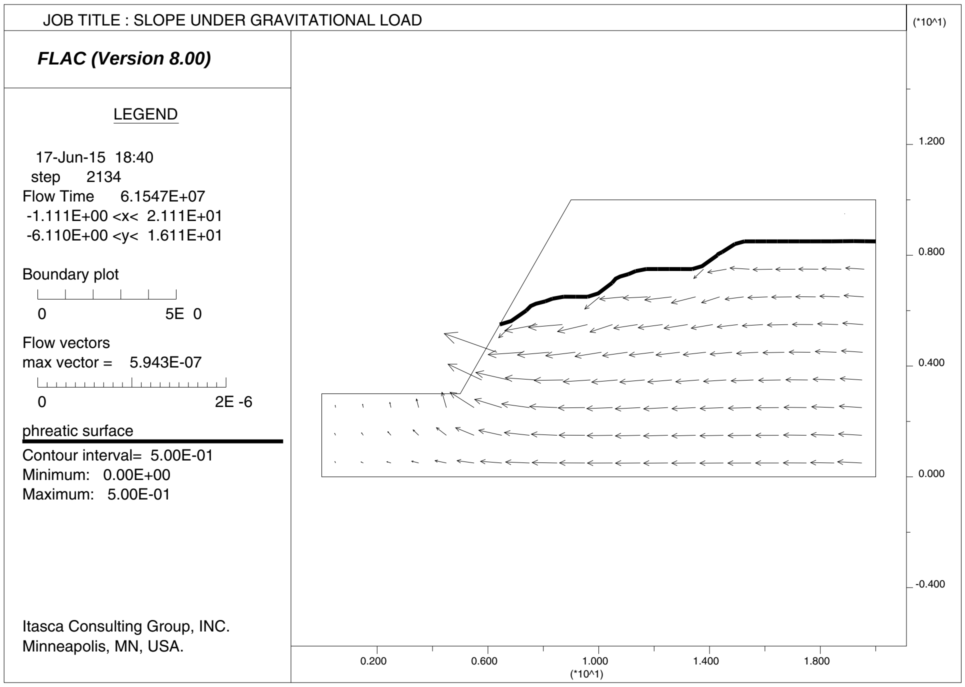

Figure 1.6 Steady-state flow through slope

Mechanical equilibrium is then established including the pore pressure. This is accomplished with the commands

```
set flow off mech on
apply press 2e4 var 0  -2e4  from 1,4  to 6,6
water bulk = 0.0
```

These commands turn off the flow calculation, turn on the mechanical calculation, apply the weight of the water to the excavation surface, and set the bulk modulus of the water to zero. The last command prevents pore pressures from generating as a result of mechanical deformation. This is done so that the results can be compared to the previous case using the water table. The **SOLVE** command is then used to find the equilibrium state.

Finally, the cohesion is reduced to 1.0×10^4, the tensile strength to zero, and the calculation is continued with the **STEP** command. The slope fails, as shown in Figure 1.7. The result is the same as that which occurs using **WATER table**.

The approach using the groundwater flow mode can take longer to reach a solution because of the extra calculation needed to establish the pore pressure distribution. However, this method is helpful when the pore pressure distribution or phreatic surface location is unknown. Also, this approach avoids the necessity of assigning wet density values to zones beneath the phreatic surface, which can become difficult for complex geometries.

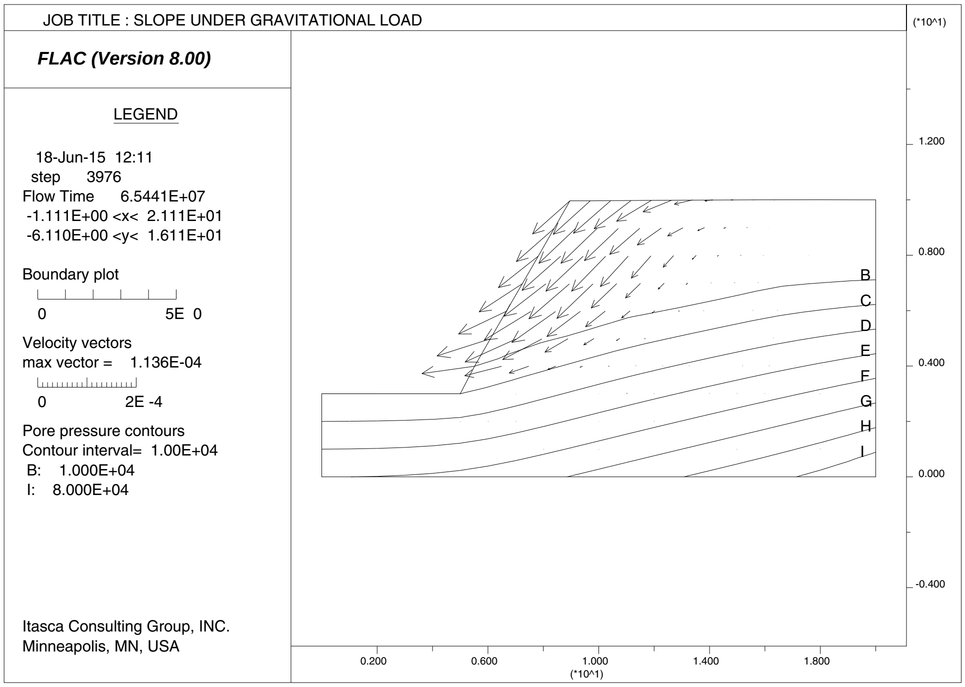

Figure 1.7 Pore pressure distribution and velocity vectors in failed slope (using **CONFIG gwflow***)*