3 PROBLEM SOLVING WITH *UDEC*

This section provides guidance in the use of *UDEC* in problem solving for rock mechanics engineering.*

In Section 3.1, an outline of the steps recommended for performing a geomechanics analysis is given, followed in Sections 3.2 through 3.10 by an examination of specific aspects that must be considered in any model creation and solution, including:

- model generation (Section 3.2);
- choice of rigid or deformable block analysis (Section 3.3);
- boundary and initial conditions (Sections 3.4 and 3.5);
- loading and sequential modeling (Section 3.6);
- choice of block and joint constitutive models and material properties (Section 3.7 and Section 3.8);
- ways to improve modeling efficiency (Section 3.9); and
- interpretation of results (Section 3.10).

You will note that *FISH* is used in this section to assist with model generation and problem solving. If you have not used *FISH* before, we recommend that you first read the *FISH* tutorial in Section 4.

Finally, the philosophy of modeling in the field of geomechanics is examined in Section 3.11; the novice modeler in this field may wish to consult this section first. The methodology of modeling in geomechanics can be significantly different from that in other engineering fields, such as structural engineering. It is important to keep this in mind when performing any geomechanics analysis.

The discussion and examples provided in this section use the command-line approach to input data to *UDEC*. Although all of the operations can be achieved by means of the graphical interface (the *GIIC* – see Section 2.2.1), the principles of model setup, execution and interpretation remain the same. *.

---

* Problem solving for coupled mechanical-hydraulic analysis involving flow in fractures and problem solving for coupled mechanical-thermal analysis are discussed in **Special Features** in Section 2 and in Section 3, respectively.

* The data files in this section are stored in the folder Datafiles\Users_Guide\Problem_Solving with the extension ".DAT." A project file is also provided for each example. For the *GIIC*, open the project file by clicking on the FILE / OPEN PROJECT menu item and select the project file name (with ".PRJ" extension). Then click on the *Project Options* icon at the top of the *Project Tree Record*, select *Rebuild unsaved states*. For the GUI, open the project file by clicking on the FILE / OPEN PROJECT menu item and select the project file name (with ".UDPRJ" extension). Then click on the *Project* tab and select the "Master.dat" and run it

3.1 General Approach

The modeling of geo-engineering processes involves special considerations, and a design philosophy different from that followed for design with fabricated materials. Analyses and designs for structures and excavations in or on rocks and soils must be achieved with relatively little site-specific data, and an awareness that deformability and strength properties may vary considerably. It is impossible to obtain complete field data at a rock or soil site. For example, information on stresses, properties and discontinuities can only be partially known, at best.

Since the input data necessary for design predictions are limited, a numerical model in geomechanics should be used primarily to understand the dominant mechanisms affecting the behavior of the system. Once the behavior of the system is understood, it is then appropriate to develop simple calculations for a design process.

This approach is oriented toward geotechnical engineering, in which there is invariably a lack of good data; but in other applications, it may be possible to use *UDEC* directly in design if sufficient data, as well as an understanding of material behavior, are available. The results produced in a *UDEC* analysis will be accurate when the program is supplied with appropriate data. Modelers should recognize that there is a continuous spectrum of situations, as illustrated in Figure 3.1:

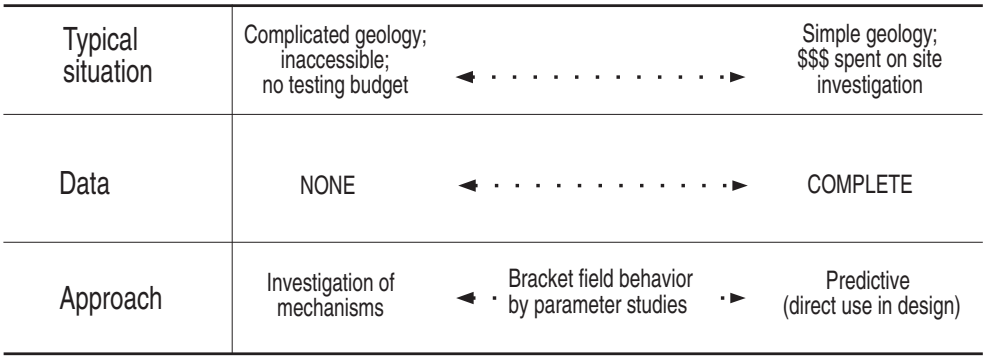

Figure 3.1 Spectrum of modeling situations

UDEC may be used either in a fully predictive mode (right-hand side of Figure 3.1) or as a "numerical laboratory" to test ideas (left-hand side). It is the field situation (and budget), rather than the program, that determines the types of use. If enough data of a high quality are available, *UDEC* can give good predictions.

Since most *UDEC* applications will be for situations in which little data are available, this section discusses the recommended approach for treating a numerical model as if it were a laboratory test. The model should never be considered as a "black box" that accepts data input at one end and produces a prediction of behavior at the other. The numerical "sample" must be prepared carefully, and several samples tested, to gain an understanding of the problem. Table 3.1 lists the steps recommended to perform a successful numerical experiment; each step is discussed separately.

Table 3.1 Recommended steps for numerical analysis in geomechanics

Step 1	Define the objectives for the model analysis
Step 2	Create a conceptual picture of the physical system
Step 3	Construct and run simple idealized models
Step 4	Assemble problem-specific data
Step 5	Prepare a series of detailed model runs
Step 6	Perform the model calculations
Step 7	Present results for interpretation

3.1.1 Step 1: Define the Objectives for the Model Analysis

The level of detail to be included in a model often depends on the purpose of the analysis. For example, if the objective is to decide between two conflicting mechanisms that are proposed to explain the behavior of a system, then a crude model may be constructed, provided that it allows the mechanisms to occur. It is tempting to include complexity in a model just because it exists in reality. However, complicating features should be omitted if they are likely to have little influence on the response of the model, or if they are irrelevant to the model's purpose. Start with a global view and add refinement as (and if) necessary.

3.1.2 Step 2: Create a Conceptual Picture of the Physical System

It is important to have a conceptual picture of the problem to provide an initial estimate of the expected behavior under the imposed conditions. Several questions should be asked when preparing this picture: Is it anticipated that the system could become unstable? Is the predominant mechanical response linear or nonlinear? Are there well-defined discontinuities that may affect the behavior, or does the material behave essentially as a continuum? Is there an influence from groundwater interaction? Is the system bounded by physical structures, or do its boundaries extend to infinity? Is there any geometric symmetry in the physical structure of the system?

These considerations will dictate the gross characteristics of the numerical model, such as the design of the model geometry, the types of material models, the boundary conditions, and the initial equilibrium state for the analysis. They will determine whether a three-dimensional model is required, or if a two-dimensional model can be used to take advantage of geometric conditions in the physical system.

3.1.3 Step 3: Construct and Run Simple Idealized Models

When idealizing a physical system for numerical analysis, it is more efficient to construct and run simple test models first, before building the detailed model. Simple models should be created at the earliest possible stage in a project to generate both data and understanding. The results can provide further insight into the conceptual picture of the system; Step 2 may need to be repeated after simple models are run.

Simple models can reveal shortcomings that can be remedied before any significant effort is invested in the analysis: Do the selected material models sufficiently represent the expected behavior? Are the boundary conditions influencing the model response? The results from the simple models can also help guide the plan for data collection by identifying which parameters have the most influence on the analysis.

3.1.4 Step 4: Assemble Problem-Specific Data

The types of data required for a model analysis include

- details of the geometry (e.g., profile of underground openings, surface topography, dam profile and rock/soil structure);
- locations of geologic structure (e.g., faults, bedding planes and joint sets);
- material behavior (e.g., elastic/plastic properties and post-failure behavior);
- initial conditions (e.g., in-situ state of stress, pore pressures and saturation); and
- external loading (e.g., explosive loading and pressurized cavern).

Since, typically, there are large uncertainties associated with specific conditions (in particular, state of stress, deformability and strength properties), a reasonable range of parameters must be selected for the investigation. The results from the simple model runs (in Step 3) can often prove helpful in determining this range, and in providing insight for the design of laboratory and field experiments to collect the needed data.

3.1.5 Step 5: Prepare a Series of Detailed Model Runs

Most often, the numerical analysis will involve a series of computer simulations that include the different mechanisms under investigation, and span the range of parameters derived from the assembled database. When preparing a set of model runs for calculation, several aspects, such as the following, should be considered.

1. How much time is required to perform each model calculation? It can be difficult to obtain sufficient information to arrive at a useful conclusion if model runtimes are excessive. Consideration should be given to performing parameter variations on multiple computers to shorten the total computation time.

2. The state of the model should be saved at several intermediate stages so that the entire run does not have to be repeated for each parameter variation. For example, if the analysis involves several loading/unloading stages, the user should be able to return to any stage, change a parameter, and continue the analysis from that stage. Consideration should be given to the amount of disk space required for save files.

3. Are there a sufficient number of monitoring locations in the model to provide for a clear interpretation of model results and for comparison with physical data? It is helpful to locate several points in the model at which a record of the change of a parameter (such as displacement, velocity or stress) can be monitored during the calculation. Also, the maximum unbalanced force in the model should always be monitored to check the equilibrium or failure state at each stage of an analysis.

3.1.6 Step 6: Perform the Model Calculations

It is best to first make one or two model runs split into separate sections before launching a series of complete runs. The runs should be checked at each stage to ensure that the response is as expected. Once there is assurance that the model is performing correctly, several data files can be linked together in a project file to run a complete calculation sequence. At any time during a sequence of runs, it should be possible to interrupt the calculation, view the results, and then continue or modify the model as appropriate.

3.1.7 Step 7: Present Results for Interpretation

The final stage of problem solving is the presentation of the results for a clear interpretation of the analysis. This is best accomplished by displaying the results graphically, either directly on the computer screen or as output to a hardcopy plotting device. The graphical output should be presented in a format that can be directly compared to field measurements and observations. Plots should clearly identify regions of interest from the analysis, such as locations of calculated stress concentrations, or areas of stable movement versus unstable movement in the model. The numeric values of any variable in the model should also be readily available for more detailed interpretation by the modeler.

We recommend that these seven steps be followed to solve geo-engineering problems efficiently. The following sections describe the application of *UDEC* to meet the specific aspects of each of these steps in this modeling approach.

3.2 Model Generation

UDEC is different from conventional numerical programs in the way the model geometry is created. A single block is created first, with a size that encompasses the physical region being analyzed. Then, this block is cut into smaller blocks whose boundaries represent both geologic features and engineered structures in the model. This cutting process is termed collectively as *joint generation*; however, "joints" represent both physically real geologic structures and boundaries of man-made structures or materials that will be removed or changed during the subsequent stages of the *UDEC* analysis. In this latter case, the joints are fictitious entities, and their presence should not influence model results. The representation of fictitious joints is discussed in Section 3.2.3.

3.2.1 Fitting the UDEC Model to a Problem Region

The *UDEC* model geometry must represent the physical problem to an extent sufficient to capture the dominant mechanisms related to the geologic structure in the region of interest. The following aspects must be considered.

1. In what detail should the geologic structure (e.g., faults, joints, bedding planes, etc.) be represented?
2. How will the location of the model boundaries influence model results?
3. If deformable blocks are used, what density of zoning is required for accurate solution in the region of interest?

All three aspects determine the size of the *UDEC* model practical for analysis.

If only a few geologic features are identified (e.g., two or three intersecting faults, or widely spaced joint sets), these may be input individually to the *UDEC* model via the **block cut crack** or **block cut joint-set** command. Keep in mind, though, that *UDEC* is a two-dimensional program; three-dimensional effects are ignored except for special cases. If the geologic structure cannot be represented adequately as planar features oriented normal to the plane of analysis, then a three-dimensional analysis (with, for example, the Itasca program *3DEC*) may be required.

Individual features may be input in one of two ways. With the **crack** keyword, the endpoints of the feature are given; with the **joint-set** keyword, the dip angle and one location point along the feature are prescribed. For example, either

```
block cut crack (0,0) (10,10)
```

or

```
block cut joint-set angle 45 origin 5,5
```

can be used to define a joint oriented at a 45° dip angle with one point along the joint at (x = 5, y = 5).

In order to be recognized in a *UDEC* calculation, a joint must be continuous (i.e., it must completely split a block into two). A joint, however, can be composed of several contiguous segments at different

angles. **crack** and **joint-set** both create discontinuous joint segments. (The **split** keyword has the same format as **crack**, but will not create a discontinuous segment.) For example, a sawtooth-shaped joint can be created with the commands in Example 3.1:

Example 3.1 Sawtooth-shaped joint

```
new
block tolerance corner-round-length 0.1
block create polygon 0,0 0,10 10,10 10,0
block cut crack 0,5 2.5,6
block cut crack 2.5,6 5,5
block cut crack 5,5 7.5,6
block cut crack 7.5,6 10,5
```

The **block cut crack** commands in this example can be given in any order. Internal cracks are saved and used if intersected by subsequently entered cracks. Any internal or partially penetrating cracks remaining after the joint generation process are deleted when model execution begins (**block cycle** or **block solve** command). Internal or partial cracks are deleted automatically when blocks are made deformable (**block zone generate** command). They can also be deleted manually by using the **block joint-delete** command.

The rounding length specified for blocks can affect the joint generation locally. The minimum block edge length is defined as twice the rounding length. Consequently, joint segments may deviate to satisfy this criterion. For example, Example 3.2 shows that if a crack is specified with an endpoint located within twice the rounding length of a corner, the crack will be relocated through the corner. *UDEC* does not warn the user about any such relocations; the user should always check by plotting the blocks.

Example 3.2 Effect of rounding length on crack generation

```
new
block tolerance corner-round-length 0.2
;block tolerance corner-round-length 0.1
block create polygon 0,0 0,10 10,10 10,0
block cut crack 0.3,0 9.7,10
```

If the rounding length is reduced to 0.1, then the crack will be located where specified.

The **block tolerance minimum-edge-length** command allows the user to define a minimum block-edge length manually. With this command, the user can set a small rounding length for solution accuracy but avoid blocks with small edge lengths and, consequently, adversely high aspect ratios (see below). For example, if the commands **block tolerance minimum-edge-length 0.4** and **block tolerance corner-round-length 0.1** are specified, then block edge lengths smaller than 0.4 will not be created, and the rounding length for blocks will be 0.1. These commands *must* be given before the **block create** command.

There is a limit to the number of joints (i.e., the number of blocks) that can be represented in the model. This is related to the extent of the model region and the number of zones in the blocks (if deformable blocks are used). Practical limits based upon available computer memory are summarized in Table 2.1. These limits must be kept in mind during the joint generation phase. As a rule, it is always best to start with only a few joints and progressively add more, if warranted, to produce the expected response. The temptation to build a complex joint structure into the model should be resisted. This approach is discussed in more detail in Section 3.11.

An automatic joint generator to create sets of joints based upon physically measured parameters (i.e., joint dip, spacing, trace length and gap length) is available in *UDEC*. This generator is also accessed via the **block cut joint-set** command and is described, with examples, in Section 3.2.2. A special joint generator program is available to create Voronoi-shaped polygons for a *UDEC* model. This can be accessed with the **block cut voronoi** command; example applications are given in Section 3.2.2. Advanced users may wish to write their own joint generators. This may be accomplished by arranging for the generator program to output a list of x,y-coordinate pairs that define the endpoints of segments with a **block cut crack** command preceding each two pairs of endpoints. Alternatively, the coordinates can be given in a **table** and the joint generated with a **block cut table** command. *FISH* may be used to automate this process. This list can then be read directly into *UDEC*. Alternatively, a graphics digitizer instrument may be used to generate coordinate pairs.

Bear in mind that joints in *UDEC* are displayed as straight-line segments in the *UDEC* model; many segments may be required to fit an irregular joint structure. The modeler must decide the level at which the *UDEC* joint geometry will match the physical jointing pattern. The effect of geometric irregularity on the response of a joint can also be taken into account via the joint material model (e.g., by varying properties along the joint).

As mentioned, the model boundary must be far enough away from the region of study so that model response is not adversely influenced. Boundary effects are described in Section 3.4. In general, for the analysis of a single underground excavation, boundaries should be located roughly five excavation diameters from the excavation periphery. The appropriate distance, however, depends on the purpose of the analysis. If failure is of primary concern, then the model boundaries may be closer; if displacements are important, then the distance to the boundaries may need to be increased.

It is important to experiment with the model to assess boundary effects. Begin with a coarse model and bracket the boundary effect using fixed and free boundary conditions while changing the distance to the boundary. The resulting effect of changing the boundary can then be evaluated in terms of differences in stress or displacement calculated in the region of interest. See Section 3.4 for an example experiment on boundary effects.

Once the block cutting is complete and boundary location has been established, the next step is to try models with different densities of block zoning. The highest density of zoning should be in regions of high stress or strain gradients (e.g., in the vicinity of excavations). For greatest accuracy, the aspect ratio of zone dimensions (i.e., triangle base-to-height ratio) should also be as near unity as possible; anything above 5:1 is potentially inaccurate. And it is not advisable have large jumps in zone size between adjacent blocks. The ratio between zone areas in adjacent blocks should not exceed roughly 4:1, for reasonable accuracy.

3.2.2 Joint Generation

Two joint generators are available in *UDEC*: a statistical joint generator that creates discontinuities defined by conventional rock-joint parameters, and a Voronoi tessellation generator that creates randomly sized polygonal blocks. The statistical joint generator is accessed with the **block cut joint-set** command, and is based upon geometric properties of rock discontinuities described in terms of statistical parameters. The Voronoi generator is accessed with the **block cut voronoi** command, and subdivides a block into convex-shaped sub-blocks of random size and shape. The generators are described in the following sections.

3.2.2.1 Statistical Joint-Set Generator

The **joint-set** joint generator in *UDEC* creates a joint pattern described in terms of selected parameters used to generate the pattern. With such a description of jointing patterns, it is possible to make quantitative statements concerning the effect of particular geometric parameters on mechanical behavior. Also, in the case of a joint pattern observed in the field, these generation parameters can be determined by matching artificial patterns to the observed pattern.

A joint set is characterized by eight generation parameters: four of these control mean geometric properties, and four control the standard deviation of random fluctuation about the mean. The keywords and parameters given with the **block cut joint-set** command are

angle a_m <a_d> **gap** g_m <g_d> **spac** s_m <s_d> **trace** t_m <t_d> **origin** x_0 y_0 **ang-dev** ad_0

where ***a*** = angle of joint track to x-axis (default = 0);
g = gap length between joint segments (default = 0);
s = spacing normal to joint tracks (default = model size); and
t = trace length of joint segment (default = model size).

For each pair of values, the first entry (with subscript $_m$) is the mean value, and the second (with subscript $_d$) is the maximum random deviation from the mean (for uniform probability distribution). The parameters are illustrated in Figure 3.2.

The final two keywords, **origin** and **ang-dev**, are provided to orient a joint trace in the model region. Their parameters are:

***x*$_0$,*y*$_0$** coordinates (global axis) of the start of one joint trace. A joint will be generated starting at (x_0,y_0); additional joints will be generated to fill the region defined by the optional **range** (by default $x_0 = 0$, $y_0 = 0$)

***ad*$_0$** deviation of angle of all joints from the direction given for the joint track. (by default $ad_0 = 0$)

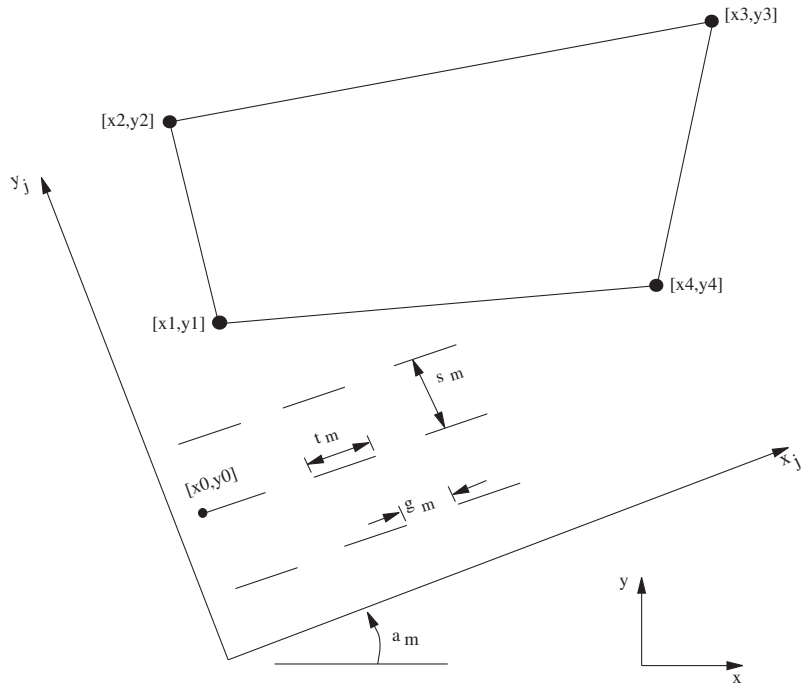

Figure 3.2 Joint set parameters

Joint generation can be limited to selected portions of the model by defining a limiting **range**. See the Help in *UDEC* for a description of the **range** keywords. In most cases, the range is defined by the **joint-region** ***n*** keyword, where ***n*** refers to the **id** number created with the **block joint-region** command.

block joint-region defines a convex quadrilateral region to limit the generation of a joint set. The parameters given with this command are:

block **jo**int-**r**egion **id** ***n x1 y1 x2 y2 x3 y3 x4 y4***

Each joint region is identified by an **id** number. The coordinates of the region are specified in a clockwise direction and delimit the boundary of the region for joint generation.

Other **range** keywords to the **join-set** keyword can also be used to restrict the blocks to be cut. If no range is specified for the **joint-set** keyword, joints will be created throughout the entire model.

The **block cut joint-set** generator may also be used to create one joint through a *UDEC* block by only specifying **angle** and **origin** to locate the joint.

Remember that **block cut joint-set** can create discontinuous joints in the *UDEC* model block. Discontinuous joints completely inside a block cannot be viewed. All discontinuous joint segments will be deleted when zone generation begins for a deformable block model, or when execution begins for a rigid block model.

3.2.2.2 Voronoi Tessellation Generator

Voronoi tessellation creates randomly sized polygonal blocks. One or more blocks in a *UDEC* model can be subdivided into Voronoi sub-blocks of arbitrary size. This joint generator is useful to simulate crack propagation; "fracturing" occurs when the joint strength between Voronoi blocks is exceeded.

The **block cut voronoi** command has the form

block **cut vo**ronoi **e**dge-**max**imim *l* <**i**terations *n*> <**r**ound *v*> <**ra**nge. . . >

An average edge length is specified for the Voronoi polygons. The polygons are randomly sized, but will have an average edge length, ***l***. The size distribution of Voronoi blocks can be made more uniform by increasing the iteration number; the default is ***n*** = 5. The rounding length can also be specified; the rounding length, ***v***, must be at least 20 times smaller than the block edge length, ***l***.

When using the **voronoi** keyword, the tessellation region will be slightly larger than the range of blocks to be subdivided; this will suppress boundary effects. The **range** keyword is used in the same manner as with the **joint-set** keyword to define the range in which the Voronoi blocks are created.

The Voronoi algorithm begins by distributing points randomly within the tessellation region. The interior points are then allowed to move. An iteration procedure moves the points; the higher the number of iterations, the more uniform the spacing between points will be. Next, triangles are created between all points. Finally, the Voronoi polygons are created by constructing perpendicular bisectors of all triangles that share a common side. The polygons are truncated at the boundaries of the tessellation region.

3.2.2.3 Examples

Several examples of joint patterns are provided to demonstrate joint generation in *UDEC*.

Example 3.3 Four regular joint sets

```
model new
block tolerance corner-round-length 0.01
block tolerance minimum-edge-length 0.02
block create polygon 0 0 0 20 20 20 20 0
block cut joint-set angle 0 spacing 3 origin 0 0
block cut joint-set angle 90 spacing 3.5 origin 0 0
block cut joint-set angle 30 spacing 4 origin 0 0
block cut joint-set angle 50 spacing 6 origin 0 0
model save 'ex3_03.sav'
return
```

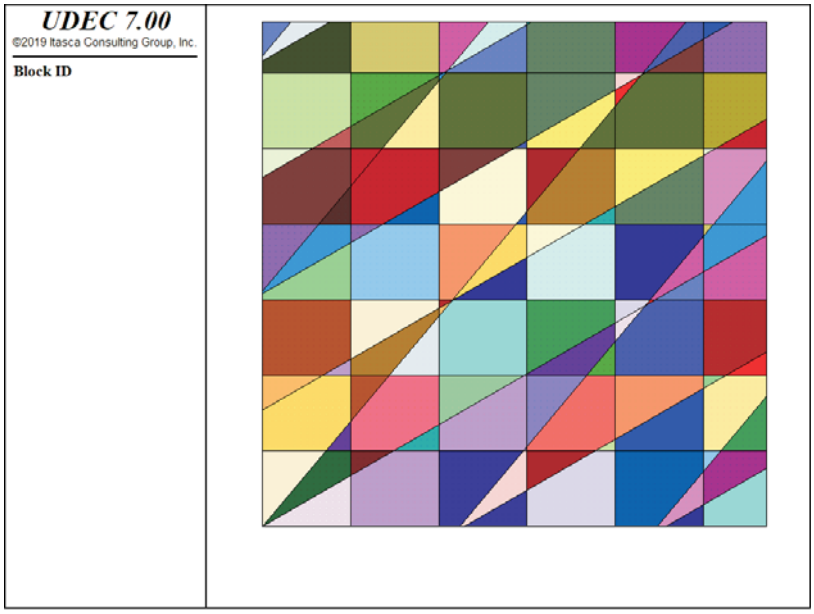

Figure 3.3 Four regular joint sets

Example 3.4 Two discontinuous joint sets

```
model new
block tolerance corner-round-length 0.01
block create circle 0.0 0.0 5.0 40
block cut joint-set angle 0 trace 3 1 gap 0.25 0.2 ...
  spacing 0.5 0.2 origin 0 0
```

```
block cut joint-set angle 70 trace 3 1 gap 0.25 0.2 ...
  spacing 0.5 0.2 origin 0 0
model save 'ex3_04.sav'
```

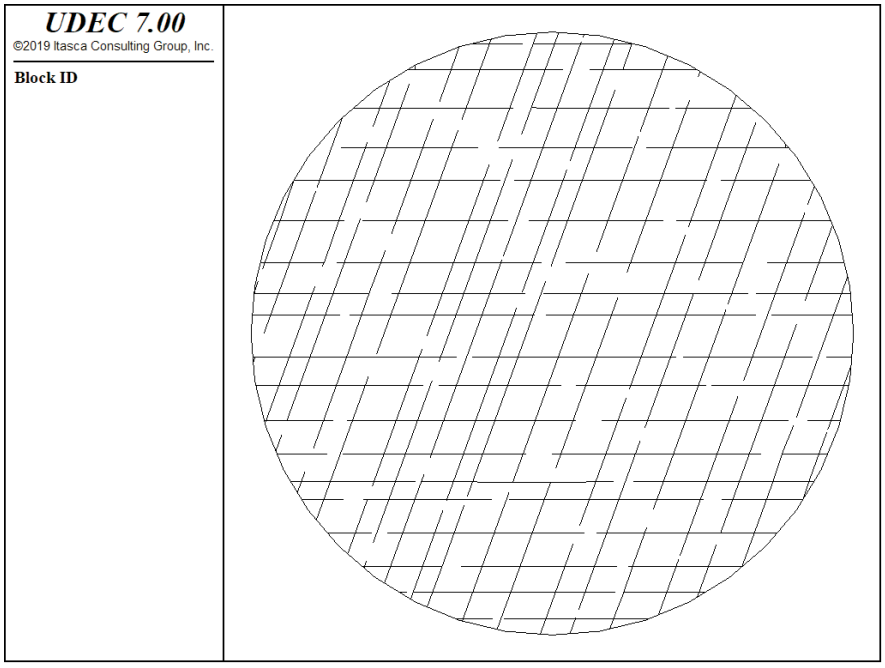

Figure 3.4 Two discontinuous joint sets

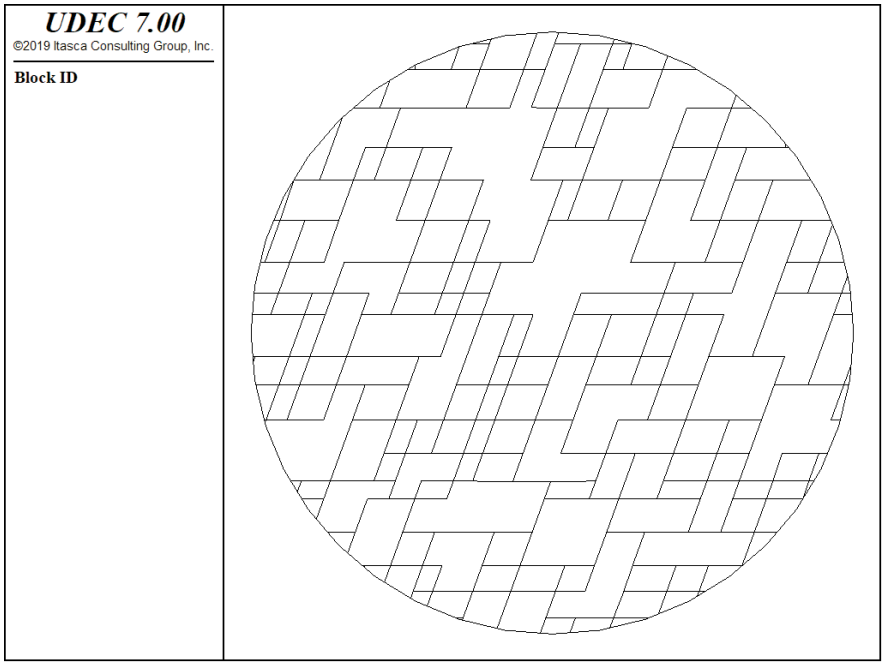

Figure 3.5 Two discontinuous joint sets with discontinuous joint segments deleted (with the **JDELETE** *command)*

Example 3.5 Blocky structure

```
model new
block tolerance corner-round-length 0.1
block create polygon 0 0 0 10 10 10 10 0
block cut joint-set angle 45 spacing 1 origin 0 0
block cut joint-set angle 315 trace 1 gap 1 spacing 1 origin 0 0
block cut joint-set angle 315 trace 1 gap 1 spacing 1 origin -1 0.354
model save 'ex3_05.sav'
```

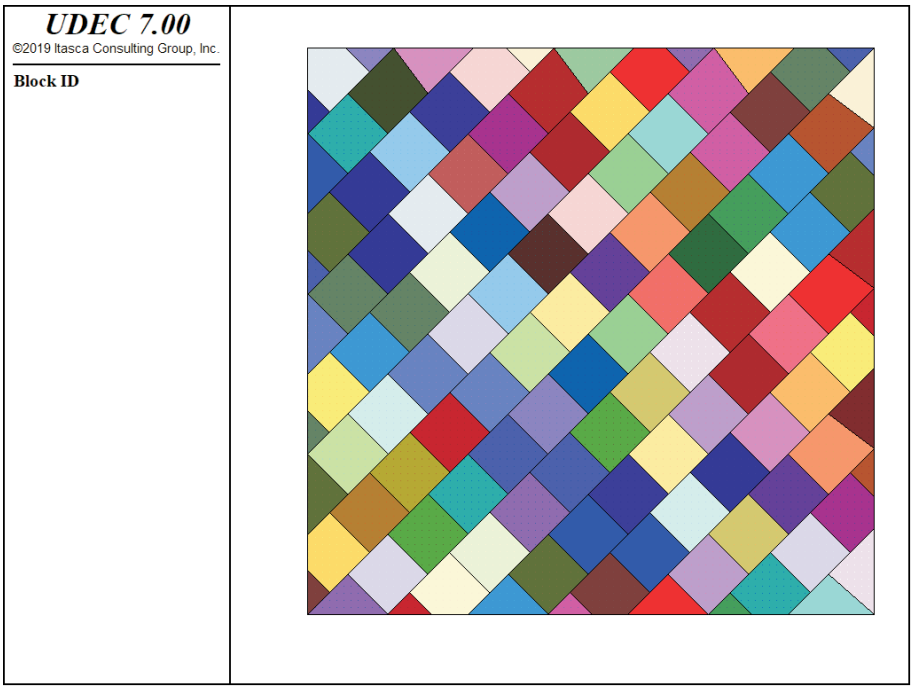

Figure 3.6 Blocky structure

Example 3.6 Two hexagonal jointing patterns

```
model new
block tolerance corner-round-length 0.01
block create polygon 0 0 0 10 10 10 10 0
block cut crack 0 5 10 5
```

```
block joint-region id 1 0.0 5.0 0.0 10.0 10.0 10.0 10.0 5.0
block joint-region id 2 0.0 0.0 0.0 5.0 10.0 5.0 10.0 0.0
; joint-region 1 joints
block cut joint-set angle 15 trace 1 gap 2 spacing 1.732 ...
  origin 1.932 0.518 range joint-region 1
block cut joint-set angle 15 trace 1 gap 2 spacing 1.732 ...
  origin 0.259 0.966 range joint-region 1
block cut joint-set angle 135 trace 1 gap 2 spacing 1.732 ...
  origin 1.932 0.518 range joint-region 1
block cut joint-set angle 255 trace 1 gap 2 spacing 1.732 ...
  origin 1.932 0.518 range joint-region 1
block cut joint-set angle 75 trace 1 gap 2 spacing 1.732 ...
  origin 0 0 range joint-region 1
block cut joint-set angle 315 trace 1 gap 2 spacing 1.732 ...
  origin 0 0 range joint-region 1
block cut joint-set angle 75 trace 1 gap 2 spacing 1.732 ...
  origin 1.225 1.225 range joint-region 1
block cut joint-set angle 135 trace 1 gap 2 spacing 1.732 ...
  origin 0.259 0.966 range joint-region 1
; joint-region 2 joints
block cut joint-set angle 0 trace 1 gap 2 spacing 1.732 ...
  origin 0 0 range joint-region 2
block cut joint-set angle 0 trace 1 gap 2 spacing 1.732 ...
  origin 1.5 0.866 range joint-region 2
block cut joint-set angle 60 trace 1 gap 2 spacing 1.732 ...
  origin 1 0 range joint-region 2
block cut joint-set angle 60 trace 1 gap 2 spacing 1.732 ...
  origin 1 1.732 range joint-region 2
block cut joint-set angle 300 trace 1 gap 2 spacing 1.732 ...
  origin 1 -1.732 range joint-region 2
block cut joint-set angle 300 trace 1 gap 2 spacing 1.732 ...
  origin 1 0 range joint-region 2
model save 'ex3_06.sav'
```

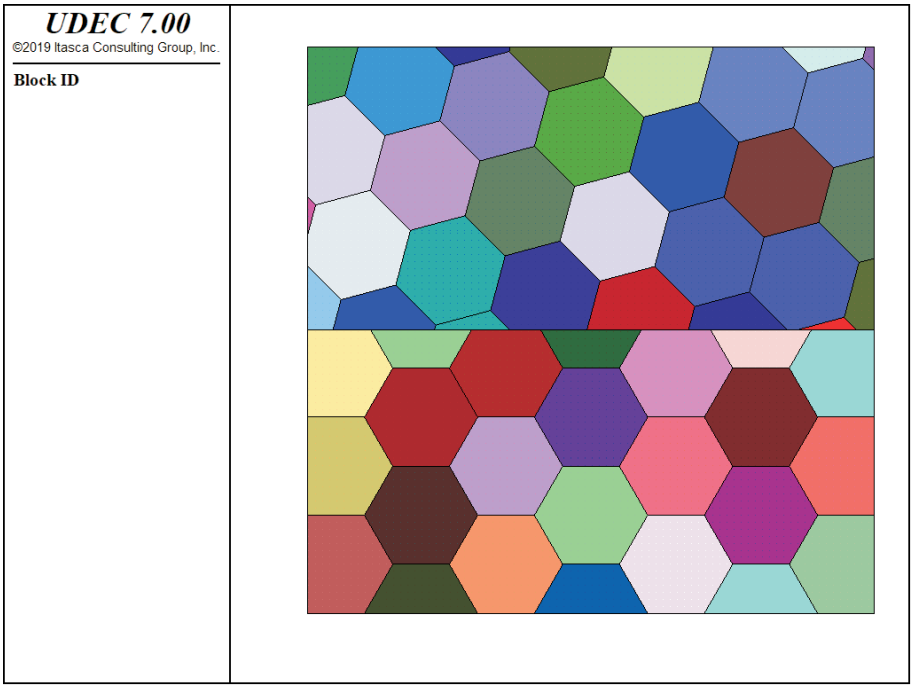

Figure 3.7 Two hexagonal jointing patterns

Example 3.7 Block embedded in Voronoi jointed system

```
model new
block tolerance corner-round-length 0.01
block create polygon 0 0 0 5 20 5 20 0
block cut crack 9 5 9 3
block cut crack 9 3 11 3
block cut crack 11 3 11 5
hide range pos-x 9 11 pos-y 3 5
block cut voronoi edge-maximum 1.0 iterations 100 round 0.01
block seek
block joint-delete
model save 'ex3_07.sav'
return
```

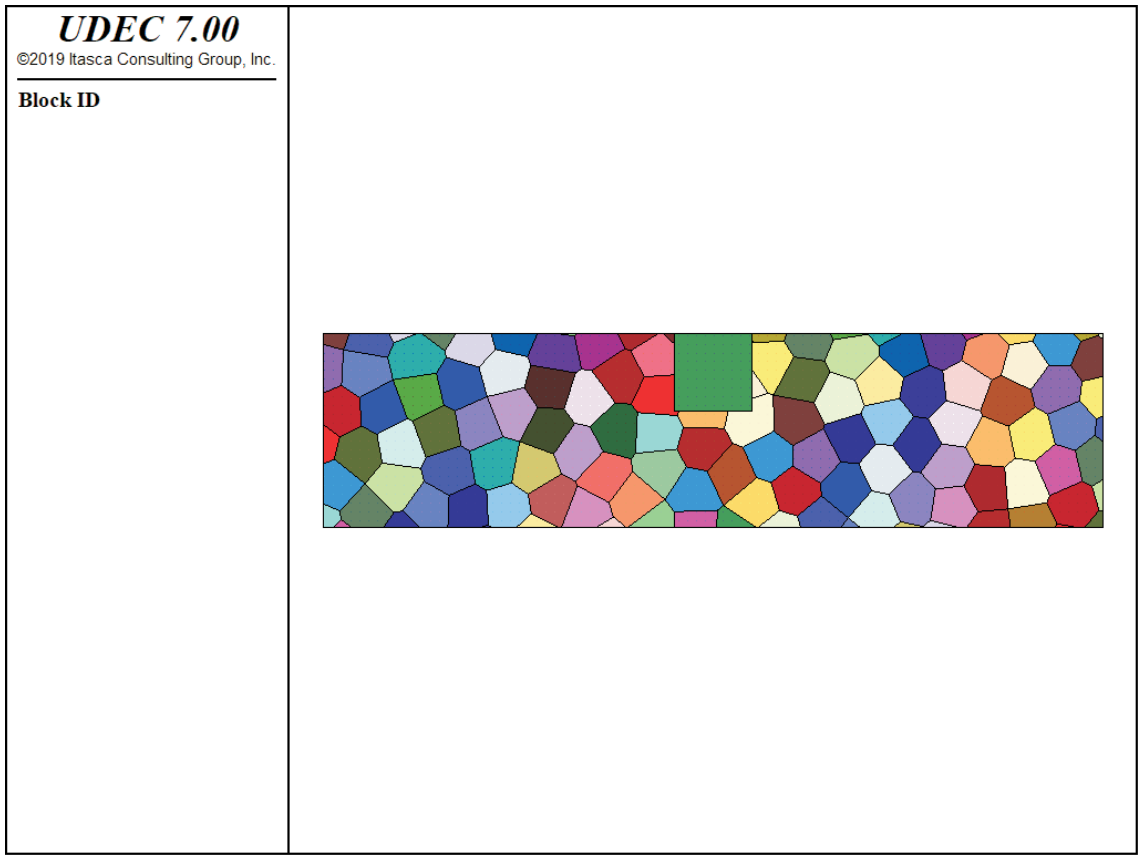

Figure 3.8 Block embedded in Voronoi jointed system

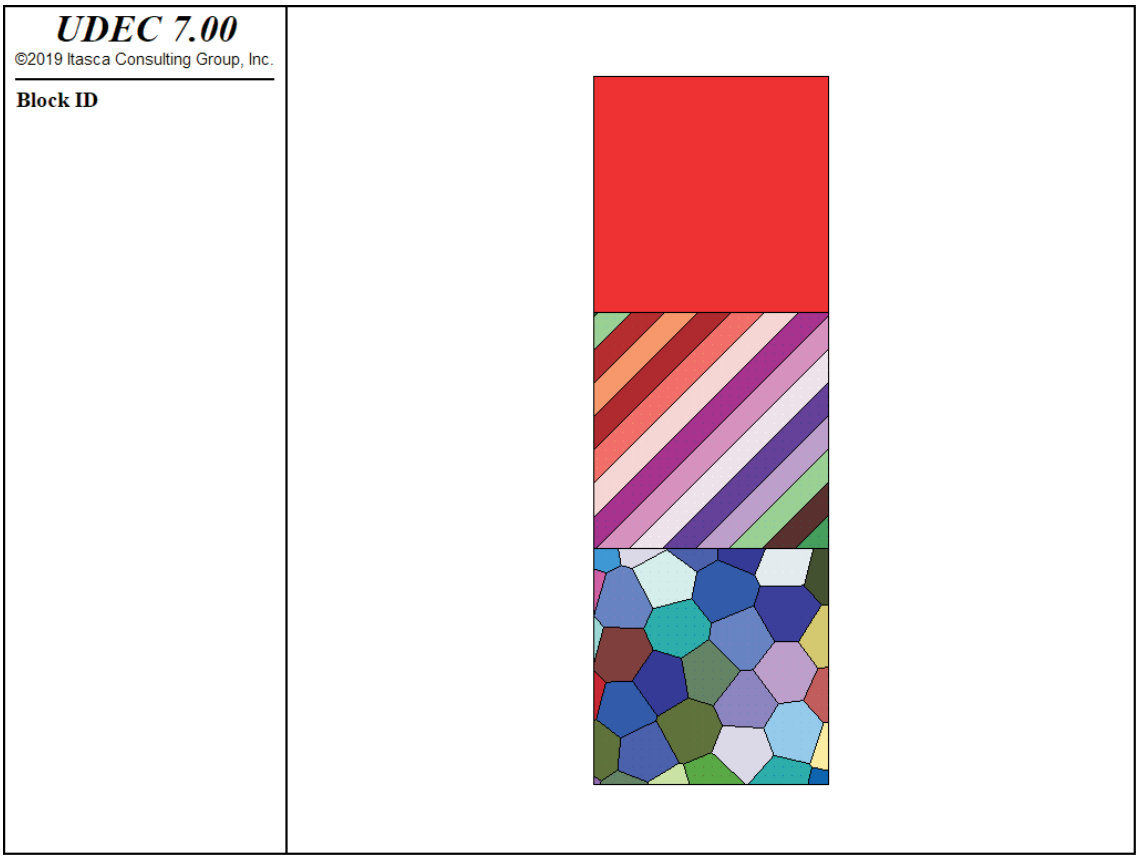

*Figure 3.9 Block embedded in Voronoi jointed system (*iteration = *100)*

***Example 3.8** Model with* joint-set *and* voronoi *joints*

```
model new
block tolerance corner-round-length 0.01
block create polygon 0 0 0 30 10 30 10 0
block cut crack 0 10 10 10
block cut crack 0 20 10 20
block hide
seek range atblock 5 5
block cut voronoi edge-maximum 2.0 iterations 100 round 0.01
block seek
block joint-delete
block joint-region id 1 0.0 10.0 0.0 20.0 10.0 20.0 10.0 10.0
block cut joint-set angle 45 spacing 1 origin 0 0 range joint-region 1
model save 'ex3_08.sav'
return
```

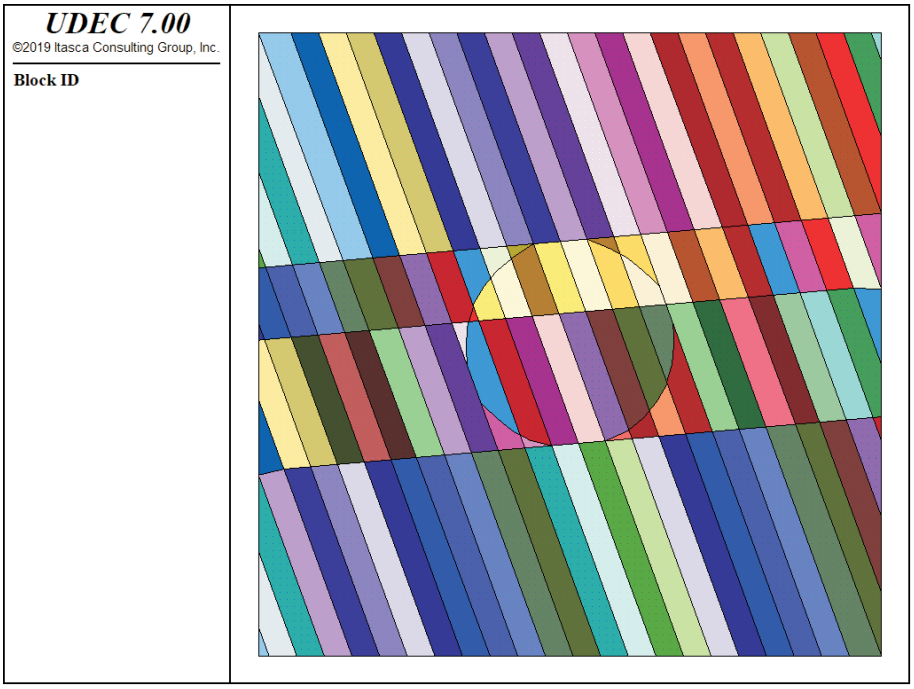

Figure 3.10 Model with* joint-set *and* voronoi *joints

3.2.3 Creating Internal Boundary Shapes

When fitting the *UDEC* model to the problem region, block boundaries must also be defined to coincide with boundary shapes of the physical problem. These may be internal boundaries representing excavations, or holes or external boundaries representing, for example, man-made structures such as earth dams or natural features such as mountain slopes. If the physical problem has a complicated boundary, it is important to assess whether simplification will have any effect on the questions that need to be answered (i.e., whether a simpler geometry will be sufficient to reproduce the important mechanisms).

All physical boundaries to be represented in the model simulation (including regions that will be added, or excavations created at a later stage in the simulation) *must* be defined before the solution process begins. Shapes of structures that will be added later in a sequential analysis must be defined and then "removed" (via **block zone cmodel assign null**) until the appropriate stage, at which point they will be activated.

The creation of boundary shapes is performed with the **block cut** keywords

arc

crack

joint-set

tunnel

Each command cuts the block into one or more segments that are fitted together in the desired shape. The **crack** and **joint-set** keywords create straight-line joint segments. The **tunnel** and **arc** keywords form straight-line segments into circular-shaped joint patterns.

In most cases, the natural features should be created in the model before the man-made features. For example, in Example 3.9 the joint structure is generated, followed by a circular tunnel excavation:

Example 3.9 Generation of a tunnel in a jointed rock

```
model new
block tolerance corner-round-length 0.05
block create polygon -6 -6 -6 6 6 6 6 -6
block cut joint-set angle 110 spacing 0.5 origin 1.85 0.77
block cut joint-set angle 5 origin 0.77 -1.85
block cut joint-set angle 5 origin 1.85 0.77
block cut joint-set angle 5 origin 0 2
block cut arc 0 0 2 0 360 32 join
model save 'ex3_09a.sav'
block delete range annulus center 0 0 radius 0 2
model save 'ex3_09b.sav'
```

The model for this problem is shown in Figure 3.11. There is one joint set dipping at 110° with a spacing of 0.50, and three individual joints, each dipping at 5°.* The circular tunnel has an origin of ($x = 0$, $y = 0$), a radius of 2 and a boundary comprising 32 joint segments.

---

* In *UDEC*, a positive dip angle is measured clockwise from the positive x-axis to the joint.

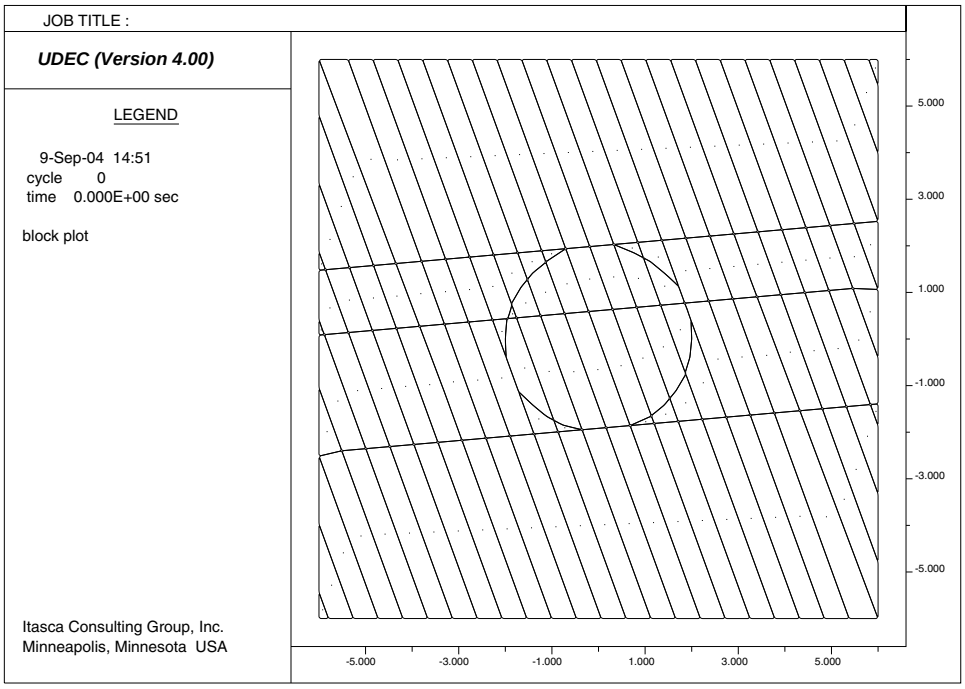

Figure 3.11 Tunnel created after joint set generation

Because the tunnel is circular, excavation of the tunnel can be performed using the **block delete range annulus** command:

```
block delete range annulus  center 0,0  radius 0,2
```

The result is shown in Figure 3.12.

Note that the joints that define the tunnel periphery are “fictitious” or “construction” joints. It is important that joints along the tunnel boundary do not influence the model response. The keyword **join** is added to the end of the **arc** keyword to designate these joints as fictitious. The **join** keyword merges gridpoints or automatically assigns high strength properties to hese joint contacts, and adjusts the joint stiffnesses such that differential joint displacements do not occur between blocks. **join** can also be specified with **crack**, **tunnel** and **joint-set** to create construction joints with these commands.

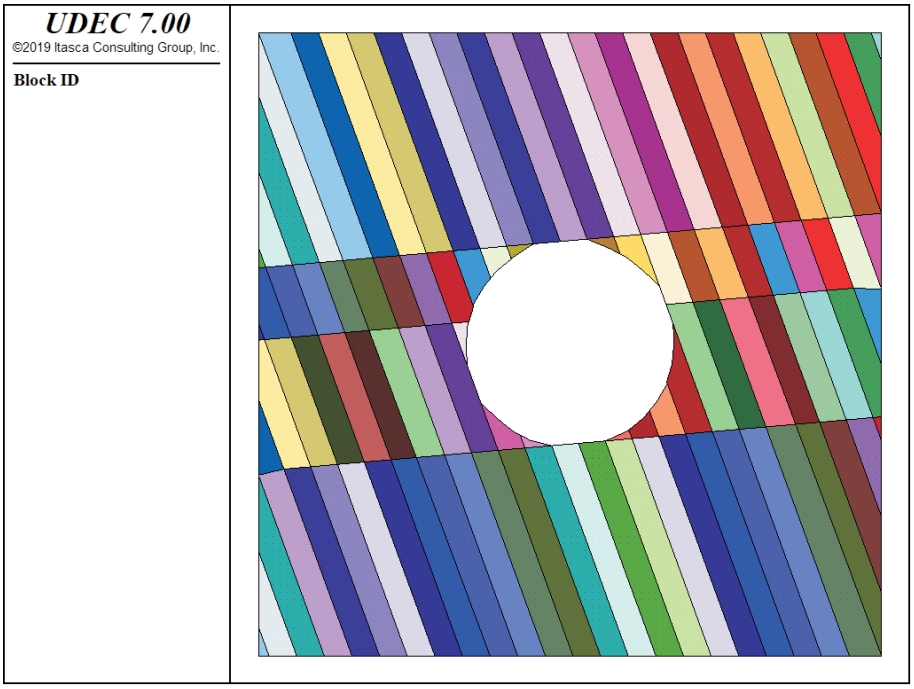

Figure 3.12 Tunnel in jointed rock

Existing joints can be made construction joints via the **block join-contact** or **block joint-block** command. Both commands can be applied with a specified **range** to limit the spatial range in which joints are joined. The **join-block** command joins all contacts between blocks whose centroids fall with the range. The **join-contact** command joins any contacts located within a given range. With **join-contact** it is possible to create internal cracks within a model region. Example 3.10 illustrates the creation of a single internal crack inside a block. A single crack is first created through the block, then two **join-contact** commands are used to join the ends of the crack. The remaining, internal crack is shown in Figure 3.13. Validation tests on a model with a single internal crack subjected to cyclic loading are given in Section 1 in the **Verification Problems**.

Example 3.10 Generation of specimen with an internal crack

```
model new
block tolerance corner-round-length 0.001
block create polygon 0 0 0 2 1 2 1 0
block cut crack 0 0.5 1 1.5
block zone gen edge 0.15
block contact join by-contact ...
  range region -0.01 0.45 -0.01 0.52 0.299 0.83 0.299 0.78
block contact join by-contact ...
  range region 0.71 1.17 0.71 1.25 1.01 1.54 1.01 1.48
block grid apply vel-y 0 ;needed to plot boundary
model sav 'ex3_10.sav'
```

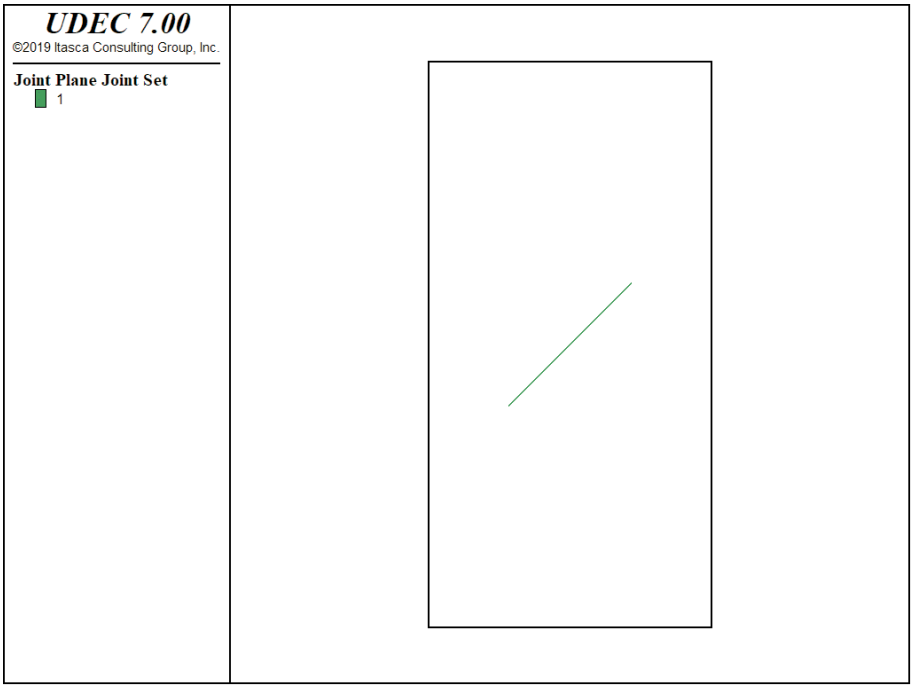

Figure 3.13 Specimen with an internal crack

Joints that are joined are assigned high values for normal and shear stiffness. It is tempting to give very high values for stiffnesses to prevent movement along a fictitious joint. However, the timestep calculation in *UDEC* is based upon stiffnesses; the response (and solution convergence) will be very slow if very high stiffnesses are specified. The lowest stiffness consistent with small joint deformation is used when joints are joined. The rule-of-thumb is that joint stiffnesses, k_n and k_s, should be set to a factor times the equivalent stiffness of the stiffest neighboring zone. The equation to calculate fictitious joint normal stiffness related to the equivalent stiffness (expressed in stress-per-distance units) of a zone in the normal direction is of the form

$$k_n = factor \times \max\left[\frac{\left(K + \frac{4}{3}G\right)}{\Delta z_{\min}}\right] \tag{3.1}$$

where $factor$ is a multiplication factor (usually set to 10)

K and G are the bulk and shear moduli, respectively; and

$\Delta z_{\min}$ is the smallest width of an adjoining zone in the normal direction (see Figure 3.14).

The max [ ] notation indicates that the maximum value over all zones adjacent to the joint is to be used (e.g., there may be several materials adjoining the joint).

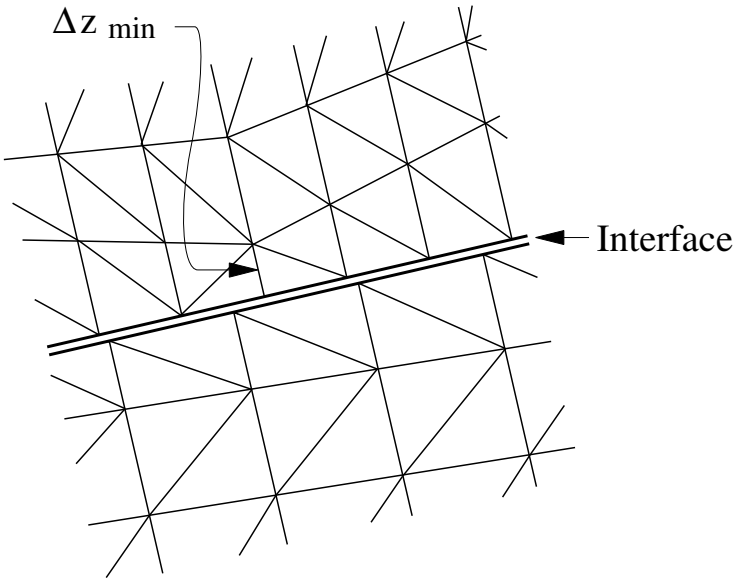

Figure 3.14 Zone dimension used in stiffness calculation

When the **join** keyword, the **block join-block** command or the **block join-contact** command is given, the joint stiffnesses are adjusted automatically following a criterion similar to Eq. (3.1).

The user can manually make a joint fictitious by applying Eq. (3.1) to specify both normal and shear stiffness. Joint properties can be changed for fictitious joints by using the **JOINT** command (see the Help in *UDEC*). For example, the joint properties for the tunnel joints in the above example can be made fictitious with the command

```
block contact prop st-n 1e11 st-s 1e11 coh 1e20 ten 1e20 ...
  range ann  0,0  1.99,2.01
```

All joints in the annular range between radii $r_1 = 1.99$ and $r_2 = 2.01$ from origin ($x = 0$, $y = 0$) will have their joint normal and shear stiffness values, and joint cohesion and tensile strength values changed.

There is one final point concerning model generation. When using continuum programs, it is usually appropriate to take advantage of symmetry conditions with excavation shapes in order to reduce the size of the model. Symmetry conditions cannot be imposed as easily with discontinuum programs, because the presence of discontinuous features precludes symmetry except for special cases. For example, it is not possible to impose a vertical line of symmetry through the model shown in Figure 3.11, because the joints in the model are not aligned with the vertical axis. It is for this reason that the axisymmetric geometry option in *UDEC* (**block config axi**) should be used with caution. This option implies that non-horizontal joints have a cylindrical or cone shape.

3.3 Selection of Deformable versus Rigid Blocks

An important aspect of a discontinuum analysis is the decision to use rigid blocks or deformable blocks to represent the behavior of intact material. The considerations for rigid versus deformable blocks are discussed in this section. If a deformable block analysis is required, there are several different models available to simulate block deformability; these are discussed in Section 3.7.

As mentioned in Section 1 in **Theory and Background**, early distinct element codes assumed that blocks were rigid. However, the importance of including block deformability is now recognized, particularly for stability analyses of underground openings and studies of seismic response of buried structures. One of the most obvious reasons to include block deformability in a distinct element analysis is the requirement to represent the "Poisson's ratio effect" of a confined rock mass.

Rock mechanics problems are usually very sensitive to the Poisson's ratio chosen for a rock mass. This is because joints and intact rock are pressure-sensitive: their failure criteria are functions of the confining stress (e.g., the Mohr-Coulomb criterion). Capturing the true Poisson behavior of a jointed rock mass is critical for meaningful numerical modeling.

The effective Poisson's ratio of a rock mass is made up of two parts: (1) a component due to the jointing; and (2) a component due to the elastic properties of the intact rock. Except at shallow depths or low confining stress levels, the compressibility of the intact rock makes a large contribution to the compressibility of a rock mass as a whole. Thus, the Poisson's ratio of the intact rock has a significant effect on the Poisson's ratio of a jointed rock mass.

A single Poisson's ratio, ν, is, strictly speaking, defined only for isotropic elastic materials. However, there are only a few jointing patterns which lead to isotropic elastic properties for a rock mass. Therefore, it is convenient to define a "Poisson effect" that can be used for discussion of anisotropic materials.

The Poisson effect will be defined as the ratio of horizontal-to-vertical stress when a load is applied in the vertical direction and no strain is allowed in the horizontal direction. Plane strain conditions are assumed. As an example, the Poisson effect for an isotropic elastic material is

$$\frac{\sigma_{xx}}{\sigma_{yy}} = \frac{\nu}{1-\nu} \tag{3.2}$$

Consider the Poisson effect produced by the vertical jointing pattern shown in Figure 3.15. If this jointing were modeled with rigid blocks, applying a vertical stress would produce no horizontal stress at all. This is clearly unrealistic because the horizontal stress produced by the Poisson's ratio of the intact rock is ignored.

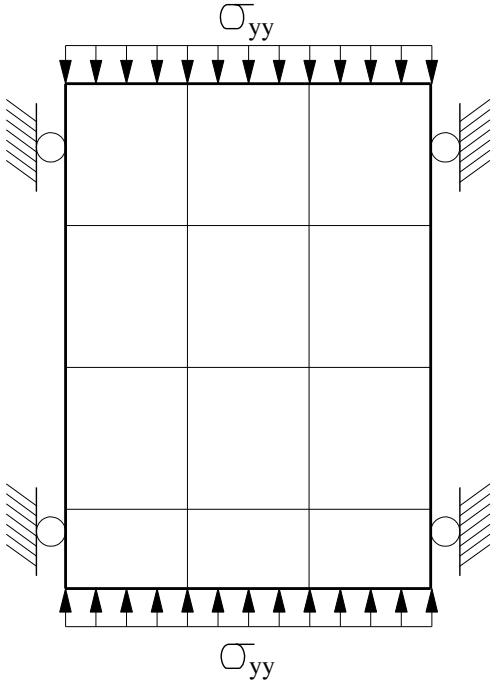

Figure 3.15 Model for Poisson effect in rock with vertical and horizontal jointing

The joints and intact rock act in series. In other words, the stresses acting on the joints and on the rock are identical. The total strain of the jointed rock mass is the sum of the strain due to the jointing and the strain due to the compressibility of the rock. The elastic properties of the rock mass as a whole can be derived by adding the compliances of the jointing and the intact rock:

$$\begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \end{bmatrix} = \left(C^{\,\text{rock}} + C^{\,\text{jointing}} \right) \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \end{bmatrix} \tag{3.3}$$

If the intact rock were modeled as an isotropic elastic material, its compliance matrix would be

$$C^{\,\text{rock}} = \frac{1+\nu}{E} \begin{bmatrix} 1-\nu & -\nu \\ -\nu & 1-\nu \end{bmatrix} \tag{3.4}$$

The compliance matrix due to the jointing is

$$C^{\,\text{jointing}} = \begin{bmatrix} \frac{1}{Sk_n} & 0 \\ 0 & \frac{1}{Sk_n} \end{bmatrix} \tag{3.5}$$

where S is the joint spacing, and k_n is the normal stiffness of the joints.

If $\epsilon_{xx} = 0$ in Eq. (3.3), then

$$\frac{\sigma_{xx}}{\sigma_{yy}} = -\frac{C_{12}^{(total)}}{C_{11}^{(total)}} \tag{3.6}$$

where $C^{(total)} = C^{(rock)} + C^{(jointing)}$.

Thus, the Poisson effect for the rock mass as a whole is

$$\frac{\sigma_{xx}}{\sigma_{yy}} = \frac{\nu\ (1+\nu)}{E/(Sk_n) + (1+\nu)(1-\nu)} \tag{3.7}$$

Eq. (3.7) is graphed as a function of the ratio $E/(Sk_n)$ in Figure 3.16. Also graphed are the results of several *UDEC* simulations run to verify the formula. The ratio $E/(Sk_n)$ is a measure of the stiffness of the intact rock in relation to the stiffness of the joints. For low values of $E/(Sk_n)$, the Poisson effect for the rock mass is dominated by the elastic properties of the intact rock. For high values of $E/(Sk_n)$, the Poisson effect is dominated by the jointing.

Now consider the Poisson effect produced by joints dipping at various angles. The Poisson effect is a function of the orientation and elastic properties of the joints. Consider the special case shown in Figure 3.17. A rock mass contains two sets of equally spaced joints dipping at an angle, θ, from the horizontal. The elastic properties of the joints consist of a normal stiffness, k_n, and a shear stiffness, k_s. The blocks of intact rock are assumed to be completely rigid.

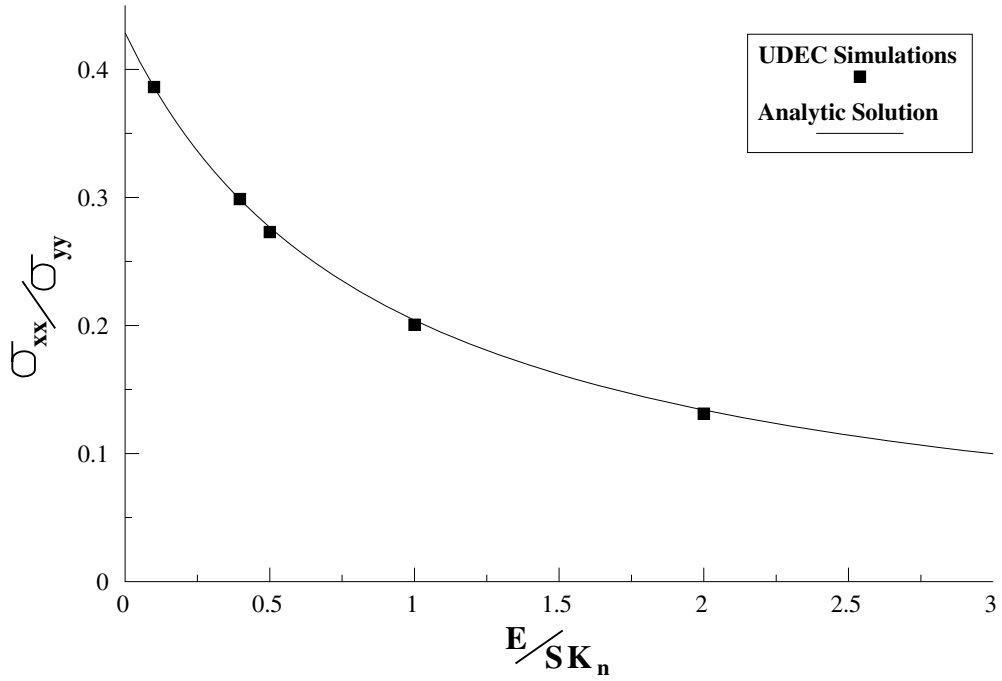

Figure 3.16 Poisson effect for vertically jointed rock (ν = 0.3 for intact rock)

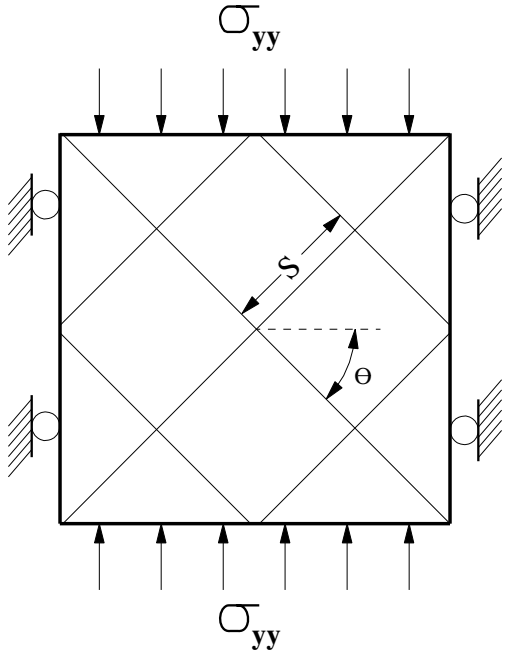

Figure 3.17 Model for Poisson effect in rock with joints dipping at angle θ from the horizontal, and with spacing, S

The Poisson effect for this jointing pattern is

$$\frac{\sigma_{xx}}{\sigma_{yy}} = \frac{\cos^2\theta\ [(k_n/k_s) - 1]}{\sin^2\theta + \cos^2\theta\ (k_n/k_s)} \tag{3.8}$$

This formula is illustrated graphically for several values of θ in Figure 3.18. Also shown are the results of numerical simulations using *UDEC*. The *UDEC* simulations agree closely with Eq. (3.8).

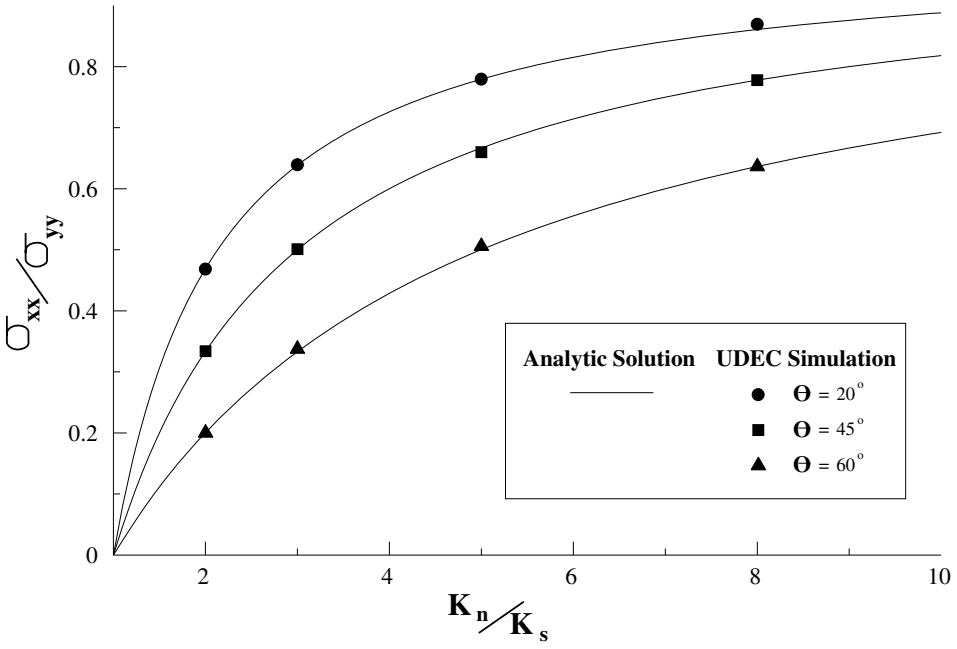

Figure 3.18 Poisson effect for jointed rock at various joint angles (blocks are rigid)

Eq. (3.8) demonstrates the importance of using realistic values for joint shear stiffness in numerical models. The ratio of shear stiffness to normal stiffness dramatically affects the Poisson response of a rock mass. If shear stiffness is equal to normal stiffness, the Poisson effect is zero. For more reasonable values of k_n/k_s, from 2.0 to 10.0, the Poisson effect is quite high, up to 0.9.

Next, the contribution of the elastic properties of the intact rock will be examined for the case of $\theta = 45°$. Following the analysis for the vertical jointing case, the intact rock will be treated as an isotropic elastic material. The elastic properties of the rock mass as a whole will be derived by adding the compliances of the jointing and the intact rock.

The compliance matrix due to the two equally spaced sets of joints dipping at 45° is

$$C^{(jointing)} = \frac{1}{2S\,k_n\,k_s} \begin{bmatrix} k_s + k_n & k_s - k_n \\ k_s - k_n & k_s + k_n \end{bmatrix}$$

Thus, the Poisson effect for the rock mass as a whole is

$$\frac{\sigma_{xx}}{\sigma_{yy}} = \frac{[\nu(1+\nu)] \,/\, E + (k_n - k_s) \,/\, (2S\,k_n k_s)}{[(1+\nu)(1-\nu)] \,/\, E + (k_n + k_s) \,/\, (2S\,k_n k_s)} \tag{3.9}$$

Eq. (3.9) is graphed for several values of the ratio $E/(Sk_n)$ in Figure 3.19 for the case of $\nu = 0.2$. Also plotted are the results of *UDEC* simulations. For low values of $E/(Sk_n)$, the Poisson effect of a rock mass is dominated by the elastic properties of the intact rock. For high values of $E/(Sk_n)$, the Poisson effect is dominated by the jointing.

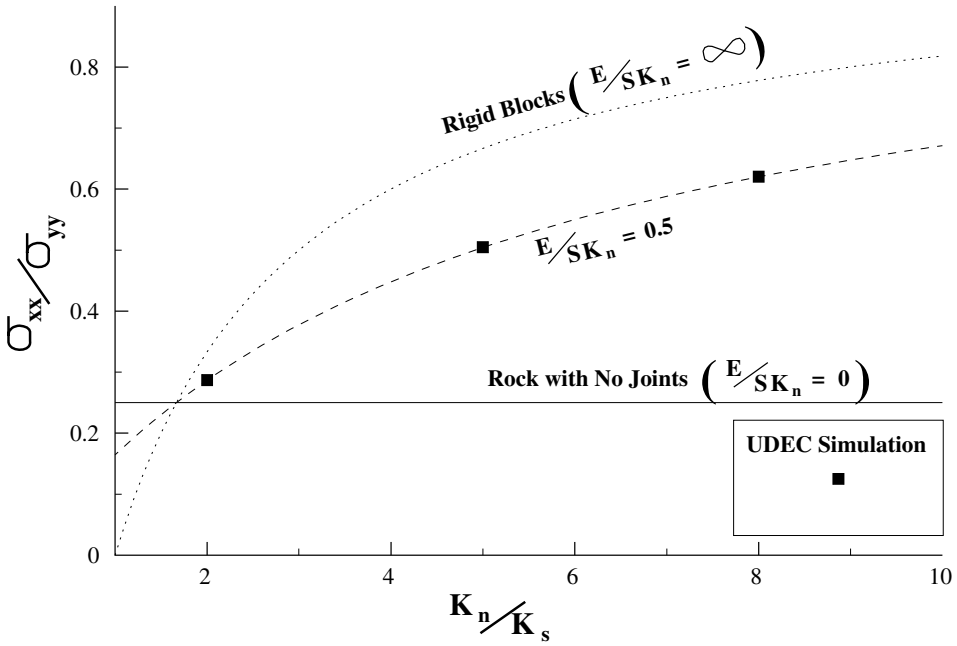

Figure 3.19 Poisson effect for rock with two equally spaced joint sets, with θ = 45° (blocks are deformable with ν = 0.2)

3.4 Boundary Conditions

The boundary conditions in a numerical model consist of the values of field variables (e.g., stress and displacement) that are prescribed at the boundary of the model. Boundaries are of two categories: real and artificial. Real boundaries exist in the physical object being modeled (e.g., a tunnel surface or the ground surface). Artificial boundaries do not exist in reality, but they must be introduced in order to enclose the chosen number of elements (i.e., blocks). The conditions that can be imposed on each type are similar; these conditions are discussed first. Then (in Section 3.4.4), some suggestions are made concerning the location and choice of artificial boundaries and the effect they have on the solution.

Mechanical boundaries are of two main types: prescribed displacement or prescribed stress. A free surface is a special case of the prescribed-stress boundary. The two types of mechanical boundary are described in Sections 3.4.1 and 3.4.2. There is a third type, the "boundary element" boundary (discussed in Section 3.4.4), which covers artificial boundaries. Viscous boundaries and free-field boundaries that are used for dynamic analysis are described in Section 4 in **Special Features**.

3.4.1 Stress Boundary

By default, the boundaries of a *UDEC* model are free of stress and any constraint. Forces or stresses may be applied to any boundary, or part of a boundary, by means of the **bock edge apply** command. Individual components of the in-plane stress tensor (σ_{xx}, σ_{xy} and σ_{yy}) are specified with the **stress** keyword. For example, the command

```
block edge apply stress 0,-1e6,-2e6   range 0,10 -1, 1
```

would apply $\sigma_{xx} = 0$, $\sigma_{xy} = -10^6$ and $\sigma_{yy} = -2 \times 10^6$ to a model boundary lying within the coordinate window $0 < x < 10$, $-1 < y < 1$. The user should always make sure that the window encompasses all the boundary corners designated for the assigned boundary condition. This can be done using the command

```
block edge list
```

Each exterior boundary corner will be listed with its assigned boundary code. The boundary can move during a model calculation, so the user must make sure that the coordinate window is large enough to include the appropriate boundary corners at the time the **block edge apply** command is executed.

Compressive stresses have a negative sign, in accordance with the general sign convention for internal stresses in *UDEC*. Also, *UDEC* actually applies stress components as forces, or *tractions*, which result from a stress tensor acting on the given boundary plane. The tractions are divided into two components: permanent and transient. Permanent tractions are constant loads, and transient tractions are time-varying loads applied for dynamic analysis by using the **history** keyword on the same command line as the **stress** keyword. Various forms of time-varying histories can be applied, including linear-varying, sine and cosine wave, and user-supplied functions; these are described in Help in *UDEC*

(see **block edge apply history**). Histories can also be applied as a *FISH* function.

Individual forces may also be applied to the model boundary by using the **force-x** and **force-y** keywords that specify x- and y-components of an applied force vector. Rigid blocks can also be assigned loads with the **force** command. Rigid block loads are applied at block centroids.

3.4.1.1 Applied Stress Gradient

The **block edge apply** command may also take the keywords **grad-x** and **grad-y**, which allow the applied stresses or forces to vary linearly over the specified range. Three parameters follow each of these keywords and describe the variation of the stress components in either the x- or y-direction:

grad-x ***sxxx sxyx syyx***
grad-y ***sxxy sxyy syyy***

The stresses vary linearly with distance from the global coordinate origin of $x = 0$, $y = 0$:

$$\begin{aligned} \sigma_{xx} &= \sigma^{\circ}_{xx} + (sxxx)x + (sxxy)y \\ \sigma_{xy} &= \sigma^{\circ}_{xy} + (sxyx)x + (sxyy)y \\ \sigma_{yy} &= \sigma^{\circ}_{yy} + (syyx)x + (syyy)y \end{aligned} \tag{3.10}$$

where σ°_{xx}, σ°_{xy} and σ°_{yy} are the stress components at the origin.

The operation of this feature is best explained by an example:

```
block edge apply stress  0,0,-10e6  grad-y 0,0,1e5   range -.1,.1 -100,0
```

The stresses at the origin ($x = 0$, $y = 0$) are

$$\sigma^{\circ}_{xx} = 0$$

$$\sigma^{\circ}_{xy} = 0$$

$$\sigma^{\circ}_{yy} = -10 \times 10^6$$

The equation for the y-variation in stress component σ_{yy} is

$$\sigma_{yy} = -10 \times 10^6 + (10^5)y$$

The value for σ_{yy} at $y = -100$ is then -20×10^6. At points in between, the y-variation is linearly scaled to the relative y-distance from the origin.

Typically, applied stress gradients are used to reproduce the effects of increasing stress with depth caused by gravity. It is important to make sure that the applied gradient is compatible with the gradient specified with the **block insitu** command and the value of gravitational acceleration (**block mech gravity** command). Section 3.5 provides more details on this matter.

3.4.1.2 Changing Boundary Stresses

As previously discussed, transient loading can be performed with the **history** keyword for dynamic analysis. For static analysis, it may also be necessary to alter the values of applied stresses during the course of a *UDEC* simulation. For example, the load on a footing may change. To effect a sudden change in an existing applied stress or load, a new **BOUNDARY** command is given with the range that encompasses the same boundary corners as in the original command, but with a change in stress value or variation.

In this case, the new value will be *added* to the existing value.* If the stress is to be removed, the current value should be given with an opposite sign. If a transient load is changed (i.e., a load assigned with the **history** keyword), any new load with the *same* history type is *added* to the existing load. However, a new transient load with a *different* history type *replaces* the old transient load.

3.4.1.3 Printing and Plotting

The boundary stresses and loads may be verified with the commands **block edge list** and **plot item create boundary**. The **block edge list** command lists the boundary corner addresses along with current values and conditions assigned to each corner. Once a **block edge apply** command is issued, a boundary corner list is created around the *entire* outer boundary of the model, regardless of the region affected by the **block edge apply** command. Optional keywords can be used with the **block edge list** command to check the different conditions along the boundary. For example,

```
block edge list force
```

lists the permanent forces (*fx*,*fy*) and incremental forces (*fxi*,*fyi*) added during the current loading stage. If transient loads are applied (with the **block edge apply**. . . **hist** command), the total forces refer to the permanent plus transient loads at the current cycle number.

---

* The user should be aware that this approach is different than the one used in the Itasca code *FLAC*, in which the stresses are *updated* to the new values when a new boundary condition is specified.

3.4.1.4 Caution and Advice

In this section, some miscellaneous difficulties with stress boundaries are described. With *UDEC*, it is possible to apply stresses to the boundary of a body that has no displacement constraints (unlike many finite element programs, which require some constraints). The body will react in exactly the same way as a real body would (i.e., if the boundary stresses are not in equilibrium, then the whole body will start moving). Example 3.11 illustrates the effect:

Example 3.11 Spin when model is not in equilibrium

```
model new
block tolerance corner-round-length 0.01
block create polygon 0 0 -3 9.54 6.54 12.54 9.54 3
block zone gen quad 2.0
block zone group 'block'
block zone cmodel assign elastic density 2.4E3 bulk 8E9 shear 5E9 ...
  range group 'block'
block edge apply stress -2000000.0 0.0 0.0 ...
  range pos-x -3.1 0.1 pos-y -0.1 9.55
block edge apply stress -2000000.0 0.0 0.0 ...
  range pos-x 6.5 9.55 pos-y 2.9 12.6
;
block cycle 500
model save 'ex3_11.sav'
```

The plot produced is given in Figure 3.20. The applied σ_{11} causes horizontal forces to act on the body. Because the body is tilted, these forces give rise to a moment that causes the body to spin.

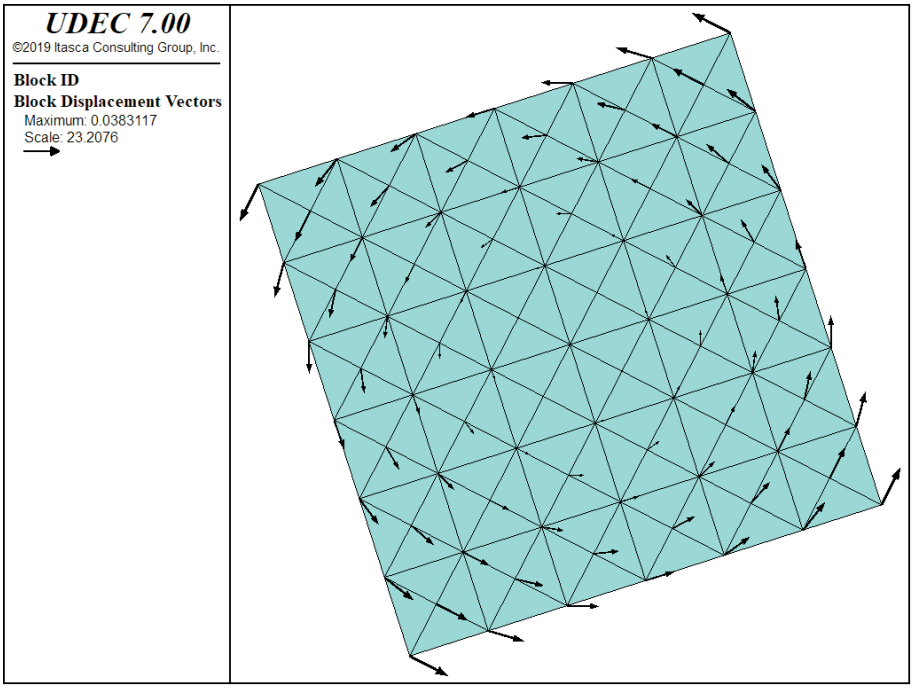

Figure 3.20 Rotational displacements induced by forces on a tilted body

A similar, but more subtle, effect arises when material is excavated from a body that is supported by a stress boundary condition: the body is initially in equilibrium under gravity, but the removal of material reduces the weight. The whole body then starts moving upward, as demonstrated in Example 3.12:

Example 3.12 Uplift when material is removed

```
model new
block tolerance corner-round-length 0.01
block create polygon 0 0 0 10 10 10 10 0
block cut crack 0 5 10 5
block zone gen quad 1.0
block zone group 'block'
block zone cmodel assign elastic density 1E3 bulk 8E9 ...
  shear 5E9 range group 'block'
block contact group 'joint'
block contact cmodel assign area stiffness-shear 1E10 ...
  stiffness-normal 1E10 range group 'joint'
; new contact default
block contact cmodel default area stiffness-shear 1E10 ...
  stiffness-normal 1E10
model gravity 0.0 -10.0
block edge apply stress 0.0 0.0 -100000.0 range pos-y -0.1 0.1
```

```
block gridpoint apply velocity-x 0 range pos-x -0.1 0.1
block gridpoint apply velocity-x 0 range pos-x 9.9 10.1
block insitu  stress 0.0 0.0 -100000.0 gradient-x 0.0 0.0 0.0 ...
  gradient-y 0.0 0.0 10000.0
block gridpoint history displacement-y 5.0 2.5
block cycle 300
block zone group 'Null:excavate' range atblock 5 7.5
block zone cmodel assign null range group 'Null:excavate'
block cycle 100
```

The difficulty encountered in running this data file can be eliminated by fixing the bottom boundary, rather than supporting it with stresses. Section 3.4.4 contains information relating to the location of such artificial boundaries.

Finally, the stress boundary affects all degrees of freedom. Velocity boundary conditions must, therefore, be prescribed *after* stress boundary conditions affecting the same boundary corners. If the stress boundary is applied after the velocity boundary, the effect of the prescribed velocity will be lost. Example 3.13 demonstrates this problem:

Example 3.13 Mixing stress and velocity boundary conditions

```
model new
block tolerance corner-round-length 0.01
block create polygon 0,0 0,10 10,10 10,0
block zone gen quad 2.0
block zone group 'block'
block zone cmodel assign elastic density 1E3 bulk 8E9 shear 5E9 ...
  range group 'block'
bl grid app velocity-y 0              range pos-y -0.1,0.1
bl edg app stress -100000.0,0.0,0.0 range pos-x -0.1,0.1
bl edg app stress -100000.0,0.0,0.0 range pos-x 9.9,10.1
bl edg app stress 0.0,0.0,-200000.0 range pos-y 9.9,10.1
block grid history disp-y 0.0,0.0
block cycle  100
ret
```

The fixed y-velocity boundary condition along the bottom boundary of the model is removed at the right and left corners when the stress boundaries are applied. These points move downward, as indicated by the history plot, when the model is loaded.

3.4.2 Displacement Boundary

Displacements cannot be controlled directly in *UDEC*; in fact, they play no part in the calculation process, as explained in Section 1 in **Theory and Background**. In order to apply a given displacement to a boundary, it is necessary to fix the boundary and prescribe the boundary's velocity for a given number of steps (using the **block gridpoint apply** command). If the desired displacement is D, a velocity, V, is applied for a time increment, T (i.e., $D = VT$), where $T = \Delta t N$, Δt is the timestep and N is the number of steps (or cycles). In practice, V should be kept small and N large, in order to minimize shocks to the system being modeled.

The **block gridpoint apply** command is used to fix the velocity of gridpoints of deformable blocks in the x- or y-direction (**bl grid apply vel-x** or **vel-y**), or in the normal or tangential direction (**vel-n** or **vel-s**) along boundaries not aligned with the x- and y-axes. The velocity of rigid blocks can be fixed with the **block fix** command (at the current velocity). The **block vel-x** causes the the velocity can be fixed at a user-selected value. The velocity can be altered with a *FISH* function.

Time-varying velocity histories can be applied via the **block grid apply** . . . **hist** command for both rigid and deformable blocks. This **history** keyword must appear on the same line as **bl grid apply vel-x** or **block grid apply vel-y** to prescribe a velocity history. Histories can also be applied as *FISH* functions. As discussed in Section 3.4.1.4, velocity boundaries should always be assigned *after* stress boundaries. Fixed velocity conditions can be removed for deformable blocks with the **block grid apply free-x** or **block grid apply free-y** command, and for rigid blocks with the **block free** command.

Velocities can also be allowed to vary linearly over a specified boundary range by adding the keyword **vel-grad** after the specified velocity keyword. Six parameters follow the **vel-grad** keyword and describe the variation of the velocity components in either the x- or y-direction:

vel-grad vx0 vy0 vxx vxy vyx vyy

The velocities vary linearly with distance from the global coordinate origin of ($x = 0$, $y = 0$):

$$\begin{aligned} vx &= vx0 + (vxx)x + (vxy)y \\ vy &= vy0 + (vyx)x + (vyy)y \end{aligned} \tag{3.11}$$

where $vx0$ and $vy0$ are the velocity components at the origin.

3.4.3 Real Boundaries – Choosing the Right Type

It is sometimes difficult to know which type of boundary condition to apply to a particular surface on the body being modeled. For example, in modeling a laboratory triaxial test, should the load applied by the platen be regarded as a stress boundary, or should the platen be treated as a rigid, displacement boundary? Of course, the whole testing machine, including the platen, could be modeled, but that might be very time-consuming. Remember that *UDEC* takes a long time to converge if there is a large contrast in stiffnesses. In general, if the object applying the load is very stiff compared with the sample (say, more than 20 times stiffer), then it may be treated as a rigid boundary. If it is soft compared with the sample (say, 20 times softer), then it may be modeled as a stress-controlled boundary. Clearly, a fluid pressure acting on the surface of a body is in the latter category. Footings on jointed rock can often be represented as rigid boundaries that move with constant velocity for the purposes of finding the collapse load of the rock. This approach has another advantage: it is much easier to control the test and obtain a good load/displacement graph. It is well-known that stiff testing machines are more stable than soft testing machines.

3.4.4 Artificial Boundaries

Artificial boundaries fall into two categories: lines of symmetry and lines of truncation.

3.4.4.1 Symmetry Lines

Sometimes it is possible to take advantage of the fact that the geometry and loading in a system are symmetrical about one or more lines. For example, if everything is symmetrical about a vertical line, then the horizontal displacements on that line will be zero. Therefore, we can make that line a boundary and fix all gridpoints in the x-direction, using the **block grid apply vel-x = 0** command. If velocities on the line of symmetry are not already zero, they will be set to zero with this command. In the case considered, the y-component of velocity on the vertical line of symmetry is not affected; it should not be fixed. Similar considerations apply to a horizontal line of symmetry. The **block grid apply vel-n = 0** command can be used to set lines of symmetry that lie at angles to the coordinate axes.

As discussed in Section 3.2.3, the presence of discontinuities makes the application of symmetry lines more difficult. When using symmetry lines in *UDEC*, the modeler should always be careful to consider the effect of joint orientation.

3.4.4.2 Boundary Truncation

When modeling infinite bodies (e.g., tunnels underground) or very large bodies, it may not be possible to cover the whole body with blocks, due to constraints on memory and computer time. Artificial boundaries are placed sufficiently far away from the area of interest that the behavior in that area is not greatly affected. It is useful to know how far away to place these boundaries and what error might be expected in the stresses and displacements computed for the areas of interest. A series of numerical experiments was performed on models containing two tunnels in an elastic material. The model contains only construction joints used to create the two tunnels. The smallest model is shown in Figure 3.21, and the largest in Figure 3.22.

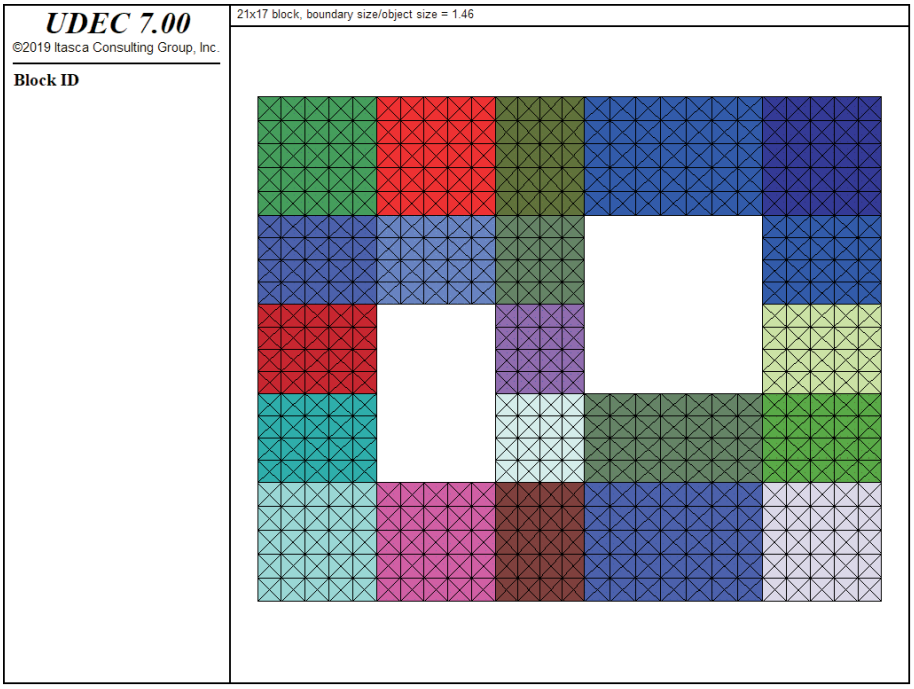

Figure 3.21 Small model with two tunnels

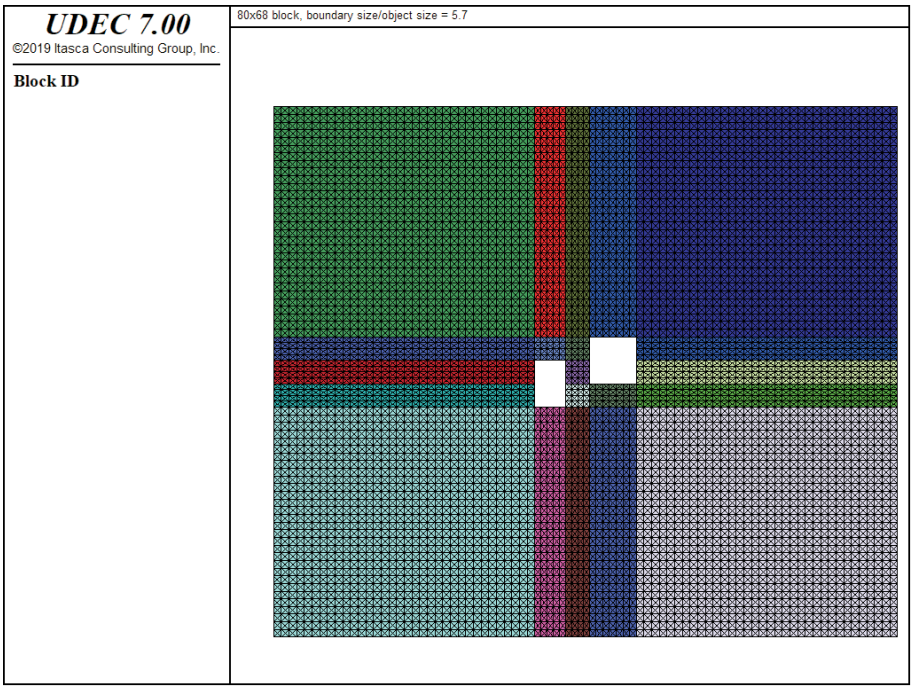

Figure 3.22 Large model with two tunnels

In all runs, the zone sizes within blocks were the same, so that discretization effects were eliminated. Two "measurements" were made: vertical displacement, u_y, at the top center point of the large tunnel; and the vertical stress, σ_{yy}, at the midpoint of the "pillar" between the two tunnels. The in-situ stress ratio was 2:1 (vertical-to-horizontal). For each rectangular boundary geometry, two runs were conducted: constant boundary stress and zero boundary displacement. In addition, runs were done with the boundary-element boundary in *UDEC*. (See Section 3.4.4.3.) Example 3.14 contains representative data files for a constant stress boundary condition, a fixed boundary condition and a boundary-element condition.

The results are summarized in nondimensional form in Figure 3.23. The measured displacements and stresses are normalized to their asymptotic values; the value of "boundary size" is the average of width and height, and the "object size" is the average distance across both tunnels. Several points may be noted from Figure 3.23:

1. A fixed boundary causes both stresses and displacements to be underestimated, while a stress boundary does the opposite.

2. The two types of boundary condition "bracket" the true solution, so that it is possible to do two tests with small boundaries and get a reasonable estimate of the true solution by averaging the two results.

3. As a rough guide, for a boundary-to-object size ratio of 5, the error in stress and displacement is around 6% for the fixed and stress boundaries.

4. The error for the closest boundary-element boundary is around 0.5%.

It may appear from these results that the boundary elements are the best way to provide an artificial boundary to simulate an infinite medium. However, there is one difficulty: the boundary element solution requires a stiffness matrix to be computed and stored. This storage can impose excessive memory overhead for larger models. (The present version of *UDEC* restricts the number of boundary-element nodes to no more than 300.) Even with this restriction, boundary elements will be the most efficient scheme for some problems. Before embarking on a series of runs, some preliminary tests, in which various types of boundary are compared, should be done.

The numerical experiments reported above are for elastic bodies. This probably represents the worst case, because the displacements and stress changes are more confined when plastic behavior is present; there is a natural cutoff distance within which most of the action occurs. The artificial boundary may be placed slightly further away without serious error. However, any artificial boundary must not be sufficiently close that it attracts plastic flow or prevents joint displacement, thereby invalidating the solution.

Example 3.14 Numerical experiments on boundary truncation

```
model new
;
;  data file for testing boundary effect
;  boundary size / object size = 1.46
;   1) stress boundary
;   2) fixed boundary
;   3) boundary-element boundary
;
block tolerance corner-round-length 0.01
block tolerance minimum-edge-length 0.02
block create polygon -10.5,-8.5 -10.5,8.5 10.5,8.5 10.5,-8.5
; smaller tunnel
block cut crack (-10.5,1.5) (10.5,1.5) join
block cut crack (-6.5,8.5) (-6.5,-8.5) join
block cut crack (-10.5,-4.5) (10.5,-4.5) join
block cut crack (-2.5,8.5) (-2.5,-8.5) join
; larger tunnel
block cut crack (10.5,4.5) (-10.5,4.5) join
block cut crack (0.5,8.5) (0.5,-8.5) join
block cut crack (10.5,-1.5) (-10.5,-1.5) join
block cut crack (6.5,-8.5) (6.5,8.5) join


block zone gen quad 1.0
block zone group 'block'
block zone cmodel assign elastic density 1850.0E-6 bulk 7.814E3 ...
  shear 4.69E3 ...
  range group 'block'
model save 'geom.sav'
;
; constant boundary stress (Sv/Sh = 2)
bl edg app stress 0.0,0.0,-20.0 range pos-y  8.4,8.6
bl edg app stress 0.0,0.0,-20.0 range pos-y -8.6,-8.4
bl edg app stress -10.0,0.0,0.0 range pos-x -10.6,-10.4
bl edg app stress -10.0,0.0,0.0 range pos-x  10.4,10.6
block insitu stress -10.0,0.0,-20.0
block solve ratio 1.0E-6

model save 'equil.sav'
block delete range pos-x -6.5,-2.5 pos-y -4.5,1.5
block delete range pos-x  0.5,6.5  pos-y -1.5,4.5
block gridpoint initial dis-x 0 dis-y 0
block gridpoint history disp-y 3.0,4.5
```

```
block zone history stress-yy -0.9,0.4
block solve ratio 1.0E-5
model save 'bstress.sav'
;
model restore 'geom.sav'
; fixed zero displacement boundary
bl grid app vel-x=0.0 vel-y=0.0 range p-x -10.6,10.6  p-y 8.4,8.6
bl grid app vel-x=0.0 vel-y=0.0 range p-x -10.6,10.6  p-y -8.6,-8.4
bl grid app vel-x=0.0 vel-y=0.0 range p-x -10.6,-10.4 p-y -8.6,8.6
bl grid app vel-x=0.0 vel-y=0.0 range p-x 10.4,10.6   p-y -8.6,8.6
block insitu stress -10.0,0.0,-20.0
block solve ratio 1.0E-5
model save 'equil1.sav'
;
block delete range pos-x -6.5,-2.5 pos-y -4.5,1.5
block delete range pos-x 0.5,6.5   pos-y -1.5,4.5
block mech reset  disp
block gridpoint history disp-y 3.0,4.5
block zone history stress-yy -0.9,0.4
block solve ratio 1.0E-6
model save 'bdisp.sav'
;
;  Boundary-Element Boundary
;
model new
block tolerance corner-round-length 0.05
block tolerance minimum-edge-length 0.1
block create circle 0.0,0.0 10.0 48
;  smaller tunnel
block cut crack (-10.5,1.5) (10.5,1.5)
block cut crack (-6.5,10.5) (-6.5,-10.5)
block cut crack (-10.5,-4.5) (10.5,-4.5)
block cut crack (-2.5,10.5) (-2.5,-10.5)
;  larger tunnel
block cut crack (10.5,4.5) (-10.5,4.5)
block cut crack (0.5,10.5) (0.5,-10.5)
block cut crack (10.5,-1.5) (-10.5,-1.5)
block cut crack (6.5,-10.5) (6.5,10.5)
;  extra cracks for better zoning
block cut crack (-8.9,-10.5) (-8.9,10.5)
block cut crack (8.9,-10.5) (8.9,10.5)
block cut crack (-10.5,7.55) (10.5,7.55)
block cut crack (-10.5,-7.55) (10.5,-7.55)
block zone gen edge 1.0
block contact join by-block
block zone group 'block'
```

```
block zone cmodel assign elastic density 1850.0E-6 bulk 7.814E3 ...
  shear 4.69E3 ...
   range group 'block'
block contact prop mat 1 stiffness-normal 1e3 stiffness-shear 1e3 ...
  friction 45 cohesion 1e20 tens 1e20
model save 'geom3.sav'
;
; constant boundary stress (Sv/Sh = 2)
block edge apply stress -10.0,0.0,-20.0
block insitu stress -10.0,0.0,-20.0
block solve ratio 1.0E-6
model save 'equil4.sav'
;
block delete range pos-x -6.5,-2.5 pos-y -4.5,1.5
block delete range pos-x 0.5,6.5   pos-y -1.5,4.5
; set boundary element boundary
block b-e gen
block b-e fix 0,-50 50,0
block b-e prop dens=1850.0e-6 bulk=7.814e3 shear=4.69e3
block b-e stiff
;
block gridpoint initial dis-x 0 dis-y 0
block gridpoint history disp-y 3.0,4.5
block zone history stress-yy -0.9,0.4
block solve ratio 1.0E-6
model save 'be.sav'
```

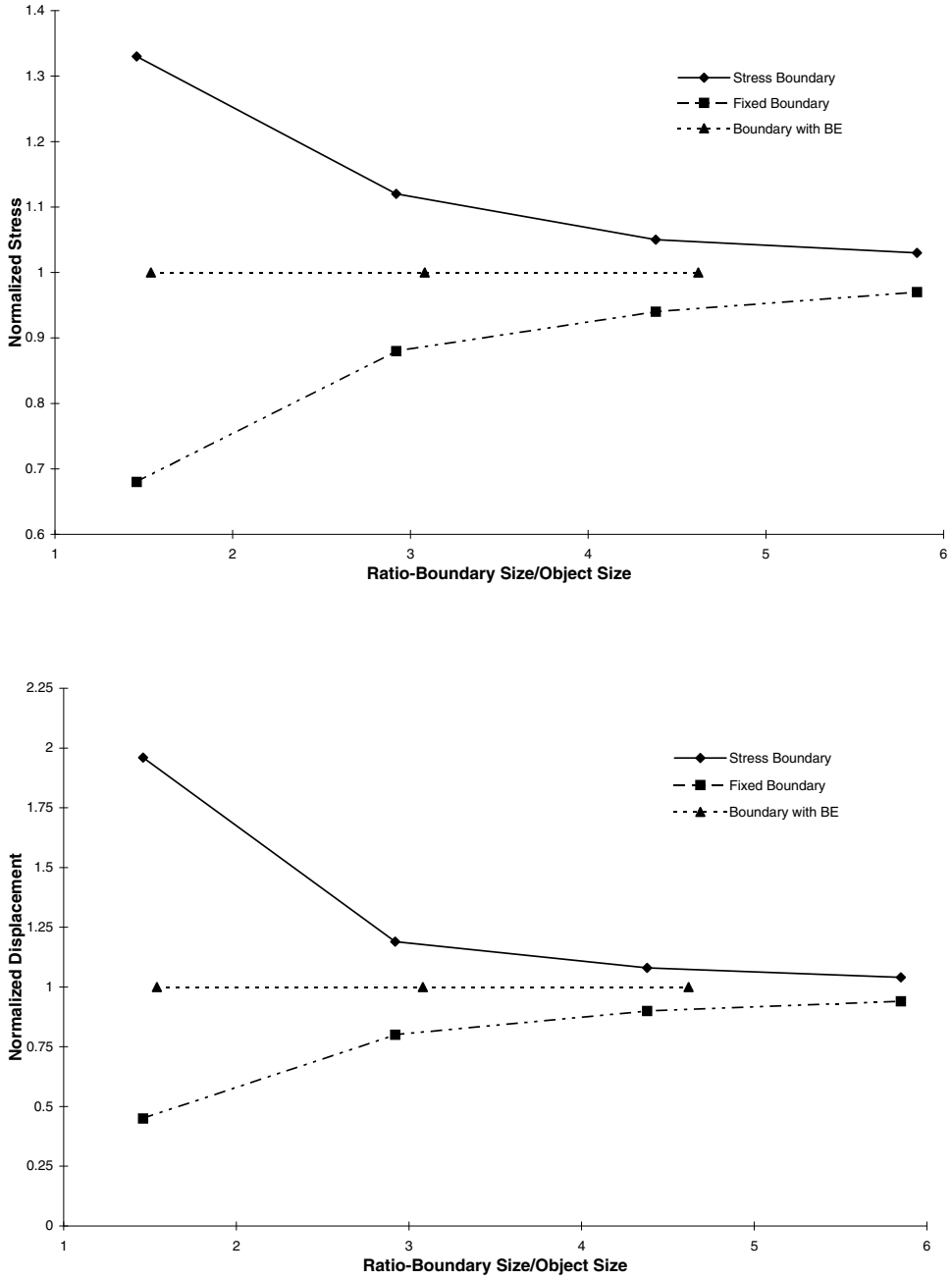

Figure 3.23 Numerical experiments on boundary truncation effects for UDEC models containing two tunnels

3.4.4.3 Boundary-Element Boundary

The boundary-element boundary condition is an artificial boundary that simulates the effect of an infinite (or semi-infinite) extent of isotropic, linear, elastic material. The formulation is a direct boundary-element solution scheme that is coupled to the distinct element scheme along the outer boundary of the *UDEC* model. The procedure used to couple the formulation is described by Lorig and Brady (1983). The distinct element region is embedded in the elastic domain. The region is first subject to in-situ initial stresses. If a change (such as an excavation) is then made in the distinct element region, the continuity of excavation-induced displacements and final state of stress must be satisfied at the distinct element-boundary element interface.

In the present formulation, boundary element nodes coincide with block corners (gridpoints) on the periphery of the distinct element domain. Determination of induced displacements at each boundary element node permits direct determination of induced nodal force by multiplying the nodal displacement vector and the stiffness matrix for the boundary element domain:

$$\bar{f} \;=\; \mathbf{K}\,\bar{u} \tag{3.12}$$

where $\mathbf{K}$ is the boundary element stiffness matrix;

$\bar{f}$ is the induced nodal force vector; and

$\bar{u}$ is the nodal displacement vector.

The induced nodal forces are related to the total nodal forces by

$$\bar{f}_t \;=\; \bar{f}_o \;+\; \bar{f} \tag{3.13}$$

where $\bar{f}_o$ represents the initial nodal force vector.

Thus, Eq. (3.13) may be written

$$\bar{f}_t \;=\; \mathbf{K}\,\bar{u} \;+\; \bar{f}_o \tag{3.14}$$

The nodal forces determined in this way can then be applied to appropriate block corners located on the periphery of the distinct element domain in the subsequent iteration step. Thus, as the relaxation process continues, nodal displacements and forces are updated and used in the subsequent calculation cycle.

Coupling between the two domains must ensure that displacements and tractions at all nodal points are continuous across the interface. These conditions require that the region of nonlinear material behavior be confined inside the distinct element domain. Surfaces of joint sliding or separation also cannot intercept the interface.

The boundary element formulation uses straight line elements, and the distinct elements are defined by straight lines, so that physical compatibility is assured. Each iteration in the distinct element

domain results in displacement and rotation of blocks. Kinematic continuity at the interface is satisfied by identifying block corner displacements with nodal displacements in the boundary element domain. Because nodes correspond to the contact of two blocks, nodal displacements are taken as the average of two adjacent corner displacements. The contact between the corners of two blocks on the interface must be elastic. Thus the difference of displacements is small, and the averaging scheme is a reasonable approximation. Because the variation of displacements is assumed linear, both in the boundary element domain and along the edge of a block, continuity of displacements across the interface is reasonably satisfied.

Each iteration results in the interface displacements that are used to compute interface forces using the boundary element stiffness matrix. If gravity is included in the distinct element domain, removal of the distinct elements from within the distinct element domain to model an excavation leads to an imbalance in forces acting at the interface. This imbalance has the effect of causing the entire problem to experience an upward rigid body motion. Convergence of the iterative process requires that these interface displacements be related to some fixed nodal point in the boundary element domain. This is accomplished by modifying the stiffness matrix so that for every set of interface displacements, a selected interior point, P, has zero displacement.

Two fixed point locations are required: one for movement in the x-direction; and one in the y-direction. The fixed points normally should be located between one and three *UDEC* model diameters from the outer boundary of the distinct element region.

The user should be aware that stresses *must* be specified in units of MPa, for the boundary element boundary to be applied. This is done to avoid problems with truncation errors in the calculation of the stiffness matrix on an Intel-based PC. Refer to Table 2.5 for consistent sets of units when assigning stress in MPa. Also, boundary elements can only be applied for the plane-strain condition, and the maximum number of boundary-element nodes is limited to 300.

See Section 7 in the **Example Applications** and Sections 5 and 7 in the **Verification Problems** for example applications of the boundary-element boundary.

3.5 Initial Conditions

In all civil or mining engineering projects, there is an in-situ state of stress in the ground, before any excavation or construction is started. By setting initial conditions in the *UDEC* model, an attempt is made to reproduce this in-situ state, because it can influence the subsequent behavior of the model. Ideally, information about the initial state comes from field measurements, but when these are not available the model can be run for a range of possible conditions. Although the range is potentially infinite, there are a number of constraining factors (e.g., the system must be in equilibrium, and the chosen yield and slip criteria must not be violated anywhere).

In a uniform layer of soil or rock with a free surface, the vertical stresses are usually equal to $g\rho z$, where g is the gravitational acceleration, ρ is the mass density of the material, and z is the depth below surface. However, the in-situ horizontal stresses are more difficult to estimate. There is a common – but erroneous – belief that there is some "natural" ratio between horizontal and vertical stress, given by $\nu/(1-\nu)$, where ν is the Poisson's ratio. This formula is derived from the assumption that gravity is suddenly applied to an elastic mass of material in which lateral movement is prevented. This condition hardly ever applies in practice, due to repeated tectonic movements, material failure, overburden removal and locked-in stresses due to faulting and localization (see Section 3.11.3). Of course, if we had enough knowledge of the history of a particular volume of material, we might simulate the whole process numerically, in order to arrive at the initial conditions for our planned engineering works. This approach is not usually feasible. Typically, we compromise: a set of stresses is installed in the model, and then *UDEC* is run until an equilibrium state is obtained. It is important to realize that there are an infinite number of equilibrium states for any given system.

In the following sections, we examine progressively more complicated situations and the ways in which the initial conditions may be specified. The user is encouraged to experiment with the various data files that are presented.

3.5.1 Uniform Stresses in an Unjointed Medium: No Gravity

For an excavation deep underground, the gravitational variation of stress from top to bottom of the excavation may be neglected because the variation is small in comparison with the magnitude of stress acting on the volume of rock to be modeled. The **block mech gravity** command may be omitted, causing the gravitational acceleration to default to zero. The initial stresses are installed with the **block insitu** command – e.g.,

```
block insitu stress  -5e6 0.0  -1e7  szz=-5e6
```

The components σ_{11} (or σ_{xx}), σ_{22} (or σ_{yy}) and σ_{33} (or σ_{zz}) are set to compressive stresses of 5×10^6, 10^7 and 5×10^6, respectively, throughout the model. Range parameters may be added if the stresses are to be restricted to a subregion of the model. It is important to remember that σ_{33} should be initialized when using *UDEC*, since all plastic constitutive models take it into account; if omitted, it defaults to zero, which may cause failure to occur in the out-of-plane direction. The **block insitu** command sets all stresses to the given values, but there is no guarantee that the stresses will be in equilibrium. There are at least two possible problems. First, the stresses may violate the yield criterion of a nonlinear constitutive model assigned to deformable blocks. In this case, plastic flow

of zones in the blocks will occur immediately after the **block cycle** command is given, and the stresses will readjust; this possibility should be checked by doing one trial step and examining the response (e.g., plotting model state). Second, the prescribed stresses at the grid boundary may not equal the given initial stresses. In this case, the boundary gridpoints will start to move as soon as a **block cycle** command is given; again, output should be checked (e.g., plotting velocity) for this possibility.

The commands in Example 3.15 produce a single block with initial stresses that are in equilibrium with prescribed boundary stresses.

Example 3.15 Initial and boundary stresses in equilibrium

```
model new
block tolerance corner-round-length 0.1
block create polygon 0 0 0 10 10 10 10 0
block zone gen quad 1.0
block zone group 'block'
block zone cmodel assign elastic density 1E3 bulk 8E9 shear 5E9 ...
  range group 'block'
block insitu  stress -5000000.0 0.0 -1.0E7 stress-ZZ -5000000.0
block edge apply stress -5000000.0 0.0 -1.0E7
block solve ratio 1.0E-5
```

3.5.2 Stresses with Gradients in an Unjointed Medium: Uniform Material

Variation in stress with depth cannot be ignored near the ground surface – the **block mech grav** command is used to inform *UDEC* that gravitational acceleration operates on the model. It is important to understand that the **block mech grav** command does not directly cause stresses to appear in the model; it simply causes *body forces* to act on all gridpoints of deformable blocks (or centroids of rigid blocks). These body forces correspond to the weight of material surrounding each gridpoint. If no initial stresses are present, the forces will cause the material to move (during stepping) in the direction of the forces until equal and opposite forces are generated by zone stresses. Given the appropriate boundary conditions (e.g., fixed bottom, roller side boundaries), the model will, in fact, generate its own gravitational stresses compatible with the applied gravity. However, this process is inefficient, since many hundreds of steps may be necessary for equilibrium. It is better to initialize the internal stresses such that they satisfy both equilibrium and the gravitational gradient. The **block insitu** command must include the **grad-x**, **grad-y** and **grad-z** parameters so that the stress gradient matches the gravitational gradient $g\rho$. The internal stresses must also match boundary stresses at stress boundaries. As mentioned in Section 3.3, there are several boundary conditions that can be used.

Consider, for example, a 20 m $\times$ 20 m box of homogeneous unjointed material at a depth of 200 m underground, with fixed base and stress boundaries on the other three sides. Example 3.16 produces an equilibrium system for this problem condition.

Example 3.16 Initial stress state with gravitational gradient

```
model new
block tolerance corner-round-length 0.02
block create polygon 0 0 0 20 20 20 20 0
block zone gen quad 2.0
block zone group 'block'
block zone cmodel assign mohr-c density 2.5E3 bulk 5E9 shear 3E9 ...
  friction 35 range group 'block'
block insitu  stress -2750000.0 0.0 -5500000.0 ...
  gradient-x 0.0 0.0 0.0 gradient-y 12500.0 0.0 25000.0 ...
  stress-ZZ -2750000.0 gradient-z 0.0 12500.0
block edge apply stress -2750000.0 0.0 -5500000.0 ...
  gradient-x 0.0 0.0 0.0 gradient-y 12500.0 0.0 25000.0 ...
range position-x -0.1 0.1 position-y 0 20
block edge apply stress -2750000.0 0.0 -5500000.0 ...
  gradient-x 0.0 0.0 0.0 gradient-y 12500.0 0.0 25000.0 ...
  range position-x 19.9 20.1 position-y 0 20
block edge apply stress 0.0 0.0 -5000000.0 ...
  range position-x 0 20 position-y 19.9 20.1
block gridpoint apply velocity-y 0 ...
  range position-x 0 20 position-y -0.1 0.1
model gravity 0.0 -10.0
block solve ratio 1.0E-5
return
```

In this example, horizontal stresses and gradients are equal to half the vertical stresses and gradients, but they may be set at any value that does not violate the yield criterion (Mohr-Coulomb, in this case). After preparing a data file such as the one above, one calculation step should be executed and the unbalanced force monitored; any failure to match internal stresses with boundary stresses will show as an unbalanced force magnitude of roughly the same order of magnitude as the applied loading. Note that the material will fail in the out-of-plane direction if **stress-zz** is omitted in the above example.

3.5.3 Stresses with Gradients in an Unjointed Medium: Nonuniform Material

It is more difficult to give the initial stresses when materials of different densities are present. Consider a layered system with a free surface, enclosed in a box with roller side boundaries and fixed base. Suppose that the material has the following density distribution:

1600 kg/m^3 from 0 to 10 m depth

2000 kg/m^3 from 10 to 15 m

2200 kg/m^3 from 15 to 25 m

An equilibrium state is produced by the data file in Example 3.17.

Example 3.17 Initial stress gradient in a nonuniform material

```
model new
block tolerance corner-round-length 0.1
block create polygon 0 0 0 25 20 25 20 0
block cut crack 0 10 20 10 join
block cut crack 0 15 20 15 join
block zone gen quad 2.0
block zone group 'mat1' range position-x 0 20 position-y 0 10
block zone group 'mat2' range position-x 0 20 position-y 10 15
block zone group 'mat3' range position-x 0 20 position-y 15 25
block zone cmodel assign elastic density 2.2E3 bulk 5E9 shear 3E9 ...
  range group 'mat1'
block zone cmodel assign elastic density 2E3 bulk 5E9 shear 3E9 ...
  range group 'mat2'
block zone cmodel assign elastic density 1.6E3 bulk 5E9 shear 3E9 ...
  range group 'mat3'
block insitu  stress 0.0 0.0 -400000.0 gradient-x 0.0 0.0 0.0 ...
  gradient-y 0.0 0.0 16000.0 range position-x 0 20 position-y 15 25
block insitu  stress 0.0 0.0 -460000.0 gradient-x 0.0 0.0 0.0 ...
  gradient-y 0.0 0.0 20000.0 range position-x 0 20 position-y 10 15
block insitu  stress 0.0 0.0 -480000.0 gradient-x 0.0 0.0 0.0 ...
  gradient-y 0.0 0.0 22000.0 range position-x 0 20 position-y 0 10
block gridpoint apply velocity-x 0 range position-x -0.1 0.1
block gridpoint apply velocity-x 0 range position-x 19.9 20.1
block gridpoint apply velocity-y 0 range position-y -0.1 0.1
model gravity 0.0 -10.0
block gridpoint history displacement-y 10.0 1.0
block gridpoint history displacement-y 10.0 12.0
block gridpoint history displacement-y 10.0 20.0
block solve ratio 1.0E-5
return
```

An individual block is created for each material density; construction joints separate each block. The internal stress profile is calculated manually for each block from the known overburden above it. Note that the example is simplified: in a real case, the elastic moduli would vary, and there would be horizontal stresses. If high horizontal stresses exist in a layer, these may also be installed with the **block insitu** command.

This example is not in equilibrium at one calculation step; approximately 40 steps are required. The presence of the fictitious construction joints prevents the model from being automatically in equilibrium even when the initial stresses match the boundary stresses. This is a result of the corner-rounding logic, which causes contact forces to be calculated at a slightly different location along block edges than the corner gridpoint forces. The larger the rounding length, the greater the

initial force imbalance. This is true for both real and fictitious joints in a model. For example, if the rounding length in the preceding data file is changed from 0.1 to 0.0001, then the model will be in equilibrium after one calculation step. The unbalanced force is nearly zero. However, in practice, the rounding length should not be set to an extremely small value, because the length also affects the creation of new contacts when motion occurs along a discontinuity. New contacts are created when the shear displacement is greater than two times the rounding length. Thus, if this length is extremely small, many new contacts may be created; this can greatly reduce the calculation speed. A small value for **block tol corner-round-length** is also more likely to induce contact overlaps that exceed the overlap limit. For practical execution times, a rounding length set to 1% of the average block edge length should be used. Some stepping, then, will usually be required to achieve equilibrium in a jointed model, whether the joints are real or fictitious, even when internal stresses are set to match boundary stresses.

3.5.4 Compaction within a Model with Nonuniform Zoning

Puzzling results are sometimes observed when a model with nonuniform zoning is allowed to come to equilibrium under gravity. A model that is composed of deformable blocks of different size will usually have nonuniform zoning. When a Mohr-Coulomb, or other nonlinear constitutive, model is assigned to the blocks, the final stress state and displacement pattern are not uniform, even though the boundaries are straight and the free surface is flat. The data file in Example 3.18 illustrates the effect (see Figure 3.24 for the generated plot showing displacement vectors and vertical stress contours).

Example 3.18 Nonuniform stress initialized in a model with nonuniform zoning

```
model new
block tolerance corner-round-length 0.1
block create polygon 0 0 0 10 10 10 10 0
block cut crack 3 0 3 10 join
block zone gen edge 1.0
block zone group 'block'
block zone cmodel assign mohr-c density 2E3 bulk 2E8 shear 1E8 ...
  friction 30 range group 'block'
block gridpoint apply velocity-x 0 range pos-x -0.1 0.1 pos-y 0 10
block gridpoint apply velocity-x 0 range pos-x 9.9 10.1 pos-y 0 10
block gridpoint apply velocity-y 0 range pos-x 0 10 pos-y -0.1 0.1
model gravity 0.0 -10.0
model save 'non0.sav'
;  SOLVE
block solve ratio 1.0E-5
model save 'non1.sav'
;  SOLVE ELASTIC
model restore 'non0.sav'
block solve ratio 1.0E-5 elastic
model save 'non2.sav'
```

```
;  INSITU STRESS
model restore 'non0.sav'
block insitu  stress -150000.0 0.0 -200000.0 ...
  gradient-x 0.0 0.0 0.0 gradient-y 15000.0 0.0 20000.0 ...
  stress-ZZ -150000.0 gradient-z 0.0 15000.0
block solve ratio 1.0E-5
model save 'non3.sav'
```

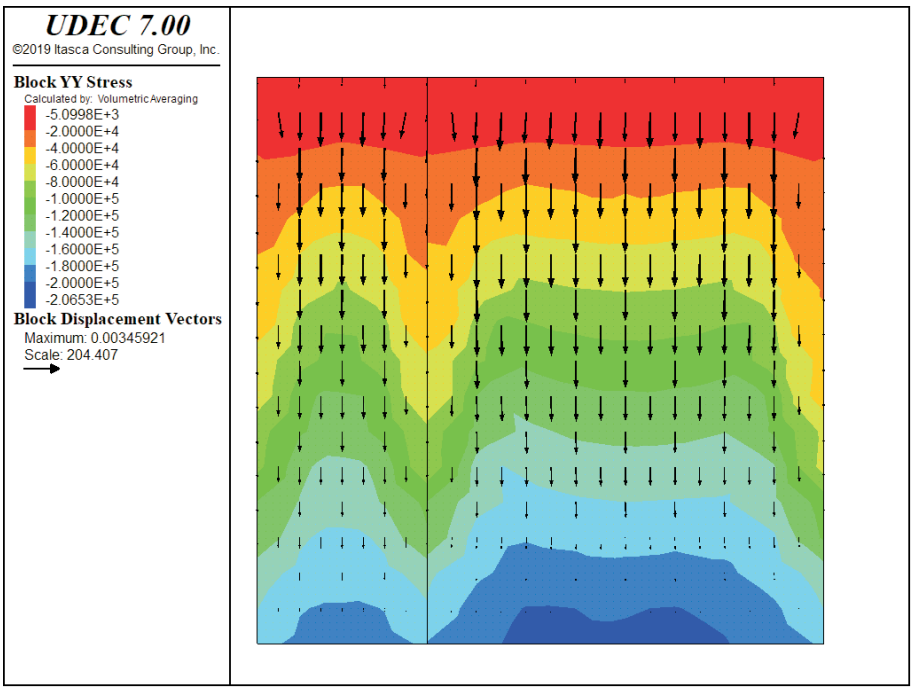

Figure 3.24 Nonuniform stresses and displacements

Since we have roller boundaries on both sides, we might expect the material to move down equally on both sides. However, the zones are smaller next to the boundary and the vertical joint than in the middle of the blocks. For static analysis, *UDEC* tries to keep the timestep equal for all zones, so it increases the inertial mass for the gridpoints at the block boundaries to compensate for the smaller zone sizes. These gridpoints then accelerate more slowly than those inside the blocks. This would have no effect on the final state of a linear material, but it causes nonuniformity in a material that is path-dependent. For a Mohr-Coulomb material without cohesion, the situation is similar to dropping sand from some height into a container and expecting the final state to be uniform. In reality, a large amount of plastic flow would occur because the confining stress does not build up immediately. Even with a uniformly zoned model, this approach is not a good one because the horizontal stresses depend on the dynamics of the process.

In order to minimize this effect of loading, the model can be run with an elastic behavior for the initial equilibrium calculation and then changed to the nonlinear behavior model for the final state. This can be done automatically by replacing the **block solve** command with **block solve elastic**.

Figure 3.25 shows the displacement vectors and vertical stress contours for this case. The material is prevented from yielding during the initial compaction process, and then the original properties are restored, and cycling is continued to the final equilibrium state.

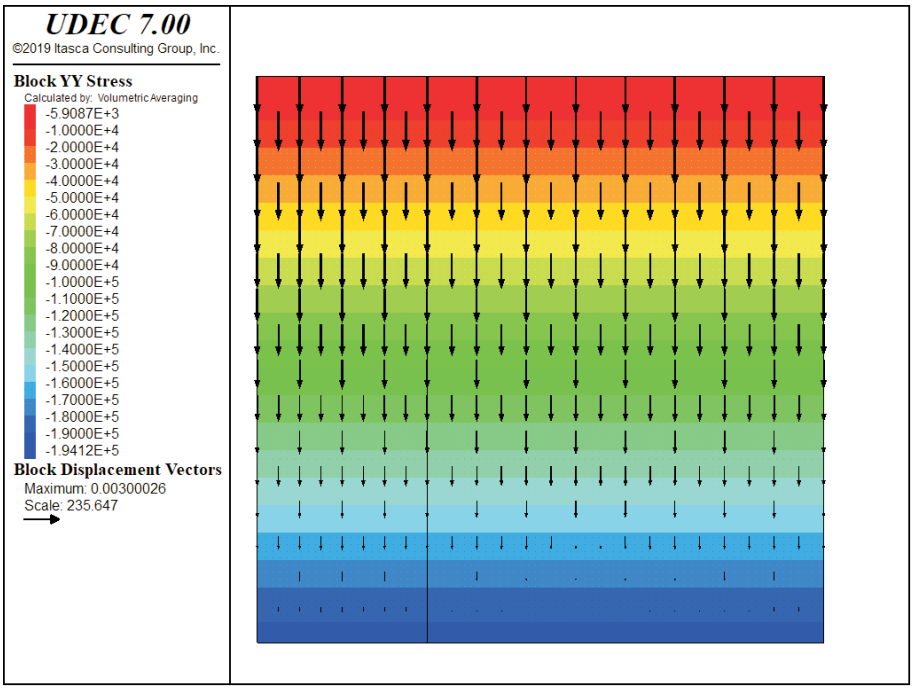

Figure 3.25 Uniform stresses and displacements

The best solution is to use the **block insitu stress** command to set initial stresses to conform to the desired K_o value (ratio of horizontal to vertical stress). For example, the following commands can be added to initialize the stress state.

```
block insitu stress (-1.5e5,0,-2.0e5)  stress-zz (-1.5e5)  &
       grad-y  (1.5e4,0,2.0e4) grad-z (0,1.5e4)
```

A stable state is achieved with $K_o = 0.75$; only a few steps are needed to reach equilibrium and the stress state is uniform.

3.5.5 Initial Stresses Following a Model Change

There may be situations in which one material model for deformable blocks is used in the process of reaching a desired stress distribution, but another model is used for the subsequent simulation. Models can be changed for entire blocks or regions of zones within blocks (via the **block zone cmodel assign** command). In either case, if one model is replaced by another non-null model, the stresses in the affected zones are preserved, as in Example 3.19:

Example 3.19 Initial stresses following a model change

```
model new
block tolerance corner-round-length 0.05
block create polygon 0 0 0 5 5 5 5 0
block zone gen quad 1.0
block zone group 'elastic block'
block zone cmodel assign elastic density 2E3 bulk 3E8 shear 2E8 ...
  range group 'elastic block'
block gridpoint apply velocity-x 0 range position-x -0.1 0.1
block gridpoint apply velocity-x 0 range position-x 4.9 5.1
block gridpoint apply velocity-y 0 range position-y -0.1 0.1
model gravity 0.0 -10.0
block solve ratio 1.0E-5
model save 'change1.sav'
;
block zone group 'mc block'
block zone cmodel assign mohr-c density 2E3 bulk 3E8 shear 2E8 ...
  friction 35 range group 'mc block'
block solve ratio 1.0E-5
model save 'change2.sav'
```

At this point in the run, the stresses generated by the initial elastic model still exist and act as initial stresses for the region containing the new Mohr-Coulomb model.

Two points should be remembered. First, if a null block or zone (**block zone cmodel assign null**) is created in any part of the model (even if it is subsequently replaced by another non-null block or zone), all stresses are removed from the null zones. Second, if one material model is replaced by another, and the stresses should physically be zero in the new model, then an **block insitu** command must be used to reset the stresses to zero in this region. This situation will occur if rock is mined out and replaced by backfill; the backfill should start its life without stress.

3.5.6 Stresses in a Jointed Medium

A spatial heterogeneity in an initial stress state can develop in a jointed and fractured medium. This results from the stress path followed during the geologic history of the medium and the physical processes, related to fracturing and slip and separation along discontinuities, which may have occurred at different stages in the history. Spatial heterogeneity of the stress state can be an important factor in the design of underground excavations, particularly if the resulting stress concentrations adversely influence the excavation stability.

It is very difficult to determine whether the stress state installed in a jointed model is representative of the in-situ state of stress. As discussed in Section 3.11.2, statistical analyses may provide a means to develop confidence in the model representation. One such study using *UDEC* is reported by Brady et al. (1986).

There are certain modeling aspects that should be considered when bringing a jointed model to an equilibrated state. First, the **block insitu** command should be invoked *after* all joints are generated in the model. Then the normal and shear stresses along joints will be initialized, corresponding to the initial stress values resolved along the plane of each joint.

As mentioned previously, a jointed model will not be initially in equilibrium even when internal stresses are set to match boundary stresses. Some calculation steps are required, and the unbalanced force should be monitored. In addition, histories of velocities or displacements at various locations in the model should be recorded. These are good indicators of the calculation step at which motion is negligible. The user should always ensure that motion in the model has essentially stopped for the equilibrium stress state before beginning the next stage of an analysis.

Joint stiffness will also affect the number of steps required to reach an equilibrium state. The rule of thumb recommended for defining stiffness values for fictitious joints (see Eq. (3.1)) is also recommended as a limiting condition for joint stiffness in general. If normal or shear stiffnesses are more than ten times higher than those defined by Eq. (3.1) for a particular problem condition, the solution time of the model will be significantly longer than that for the case in which the stiffnesses are limited to ten times the value from this equation, without a significant change in the behavior of the system. This effect is discussed in more detail in Section 3.6.

It is possible that, for the specified initial stress state and joint strength properties, some joints will slip or separate when the model is brought to an equilibrated state. Joint slip that is confined within the model is acceptable; “locked-in” stresses at the joint ends will result. However, the user should avoid conditions for which joint failure extends to the model boundary. This indicates that the model conditions are not well-posed. It may be necessary to reevaluate the assigned stress state, joint properties and joint orientations and locations. If conditions are such that joint failure still extends to a boundary, then a fixed boundary condition should be considered. This implies that the joint is truncated at the boundary.

Plot the joints with the state attribute to identify lengths of joints that have slipped or separated.

The data file in Example 3.20 demonstrates the case of a joint dipping at 60° confined between two joints dipping at 20°. The 60° joint slips for the prescribed initial stress while the 20° joints do not. The friction angle for all joints is 30°.

Example 3.20 Slip of a confined joint

```
model new
block tolerance corner-round-length 0.1
block create polygon 0 0 0 20 20 20 20 0
block cut joint-set angle 20 spacing 50 origin 10 14 join
block cut joint-set angle 20 spacing 50 origin 10 6 join
block joint-region id 1 5.0 5.0 5.0 15.0 15.0 15.0 15.0 5.0
block cut joint-set angle 60 spacing 50 origin 10 10 range joint-region 1
block zone gen quad 2.0
block zone group 'block'
block zone cmodel assign elastic density 2E3 bulk 8E9 shear 5E9 ...
  range group 'block'
block contact group 'joint'
block contact cmodel assign area stiffness-shear 5E10 ...
  stiffness-normal 2.5E10 friction 30 range group 'joint'
; new contact default
block contact cmodel default area stiffness-shear 5E10 ...
  stiffness-normal 2.5E10 friction 30
block insitu  stress -2500000.0 0.0 -1.0E7
block edge apply stress -2500000.0 0.0 0.0 range position-x -0.1 0.1
block edge apply stress -2500000.0 0.0 0.0 range position-x 19.9 20.1
block edge apply stress 0.0 0.0 -1.0E7 range position-y 19.9 20.1
block gridpoint apply velocity-y 0 range position-y -0.1 0.1
block solve ratio 1.0E-5
model save 'slip.sav'
```

Figure 3.26 shows a block plot on which the region of joint slip is indicated by a thick line. Contours of σ_{xy} are also plotted and show the areas of locked-in stresses near the ends of the 60° joint.

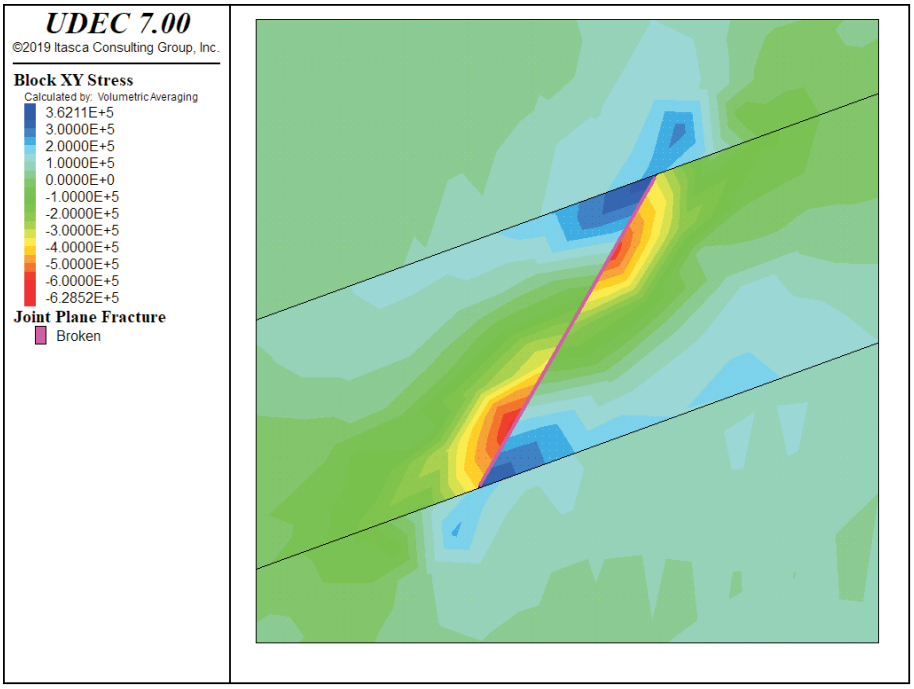

Figure 3.26 Slip of a confined joint; plot shows shear stress contours

3.6 Loading and Sequential Modeling

By applying different model loading conditions at different stages of an analysis, it is possible to simulate changes in physical loading, such as sequences of excavation and construction. Changes in loading may be specified in a number of ways (e.g., by applying new stress or displacement boundaries, by changing the material model in blocks to either a null material or to a different material model, or by changing material properties).

It is important to recognize that sequential modeling follows the stages of an engineering work. In most analyses, each work stage corresponds to a different static solution following a loading change (i.e., physical time is not a parameter). *UDEC* can perform calculations for transient flow through joints, heat transfer and dynamic mechanical analysis, as well. In these cases, a static solution for an equilibrium stress state may be followed, for example, by a dynamic calculation for an applied explosive excitation or a transient calculation for flow through joints.

Time-dependent behavior, on the other hand, cannot be simulated directly. Some engineering judgment must be used to estimate the effects of time. For example, a model parameter may be changed after a predetermined amount of displacement or strain has occurred. This displacement may be estimated to have occurred over a given period of time.

A loading change must cause unbalanced forces to develop in order to effect a change in model response. Therefore, changing the elastic properties will have no effect, whereas changing strength properties will, if the change causes the current stress state to exceed the failure limit.

The recommended approach to sequential modeling is demonstrated by the following example. This problem involves the stability analysis of an underground opening in jointed rock and includes both static and dynamic computation stages. There are three stages to be analyzed:

(1) equilibration at the in-situ stress state;

(2) excavation of the tunnel; and

(3) application of the dynamic loading.

The objective is to investigate the stability of the excavation under in-situ conditions and when subjected to a dynamic wave.

The tunnel is located in rock containing two continuous joint sets oriented at ±15° from vertical with a constant spacing of 1 m. The model is created by the following series of commands beginning with Example 3.21:

Example 3.21 Stability analysis of an underground excavation – initial model

```
new
block delete-open on
block tolerance corner-round-length 7.5E-3
block create polygon -80.0,-40.0 -80.0,80.0 80.0,80.0 80.0,-40
; boundary cracks for jointed region
block cut crack (-80,45) (80,45) join
block cut crack (-80,15) (80,15) join
block cut crack (10.34,15) (10.34,45) join
block cut crack (-10.34,15) (-10.34,45) join
; excavation boundary cracks
block cut crack (0,36.5) (1,36.5) join
block cut crack (1,36.5) (2,36) join
block cut crack (2,36) (2.5,35) join
block cut crack (2.5,35) (2.5,30) join
block cut crack (0,30) (2.5,30) join
block cut crack (0,36.5) (-1,36.5) join
block cut crack (-1,36.5) (-2,36) join
block cut crack (-2,36) (-2.5,35) join
block cut crack (-2.5,35) (-2.5,30) join
block cut crack (0,30) (-2.5,30) join
; jointed region
block joint-region id 1 -10.34,15.0 -10.34,45.0 10.34,45.0 10.34,15.0
block cut joint-set angle 75 spacing 1 origin 0,28.5 range jregion 1
block cut joint-set angle 105 spacing 1 origin 0,28.5 range jregion 1
block delete range area 0.1
; zoning

block zone gen edge 3.5 range pos-x -10.33,10.33 pos-y 15,45
```

```
block zone gen quad 4.0
; rock mass properties
block zone group 'rock'
block zone cmodel assign elastic density 2.34E3 bulk 8.39E9 ...
  shear 6.29E9  range group 'rock'
; rock joint properties
group joint 'joints'
joint model residual jks 1E10 jkn 1E10 jfriction 20 jcohesion 1E6 ...
  range group 'joints'
; new contact default
block contact cmodel default residual jks=1E10 jkn=1E10 ...
  jfriction=20 jcohesion=1E6
; boundary and initial conditions
boundary xvelocity 0 range pos-x -81,-79 pos-y -40,80
boundary xvelocity 0 range pos-x 79,81   pos-y -40,80
boundary yvelocity 0 range pos-x -80,80  pos-y -41,-39
insitu stress -1872000.0,0.0,-1872000.0 xgrad 0.0,0.0,0.0 ...
       ygrad 23400.0,0.0,23400.0
block mech gravity=0.0 -10.0
; solve for initial equilibrium
block gridpoint history displace-y 0.0,34.0
block solve ratio 1.0E-7 elastic
model save 'step1.sav'
ret
; excavate tunnel
block delete range pos-x -2.6,2.6   pos-y 30,35
block delete range pos-x -2.46,2.46 pos-y 34.6,36.5
block delete range pos-x -2.9,3.2   pos-y 31.9,32.6
block grid reset  disp
history reset
block grid history displace-y 0.0,36.2
block grid history displace-x 3.1,32.5
block solve ratio 1.0E-6
model save 'step2.sav'
```

Figures 3.27 shows the resulting model configuration. The tunnel is approximately 6.5 m in height and 6 m in width, with the tunnel invert at elevation 30 m. The model lateral boundaries are located approximately 75 m from the tunnel in order to minimize the influence of the boundaries on the response of the tunnel. The top boundary of the model is located at the ground surface (elevation 80 m). The joints are only created within a limited region around the tunnel sufficient to encompass the extent of joint failure. By restricting the joint generation, the calculation time can be reduced. Fictitious joints are assigned to the boundary joints around the tunnel region by adding the **join** keyword to the **block cut crack** commands that create these joints. A close-up view of the tunnel region with the tunnel excavated is shown in Figure 3.28.

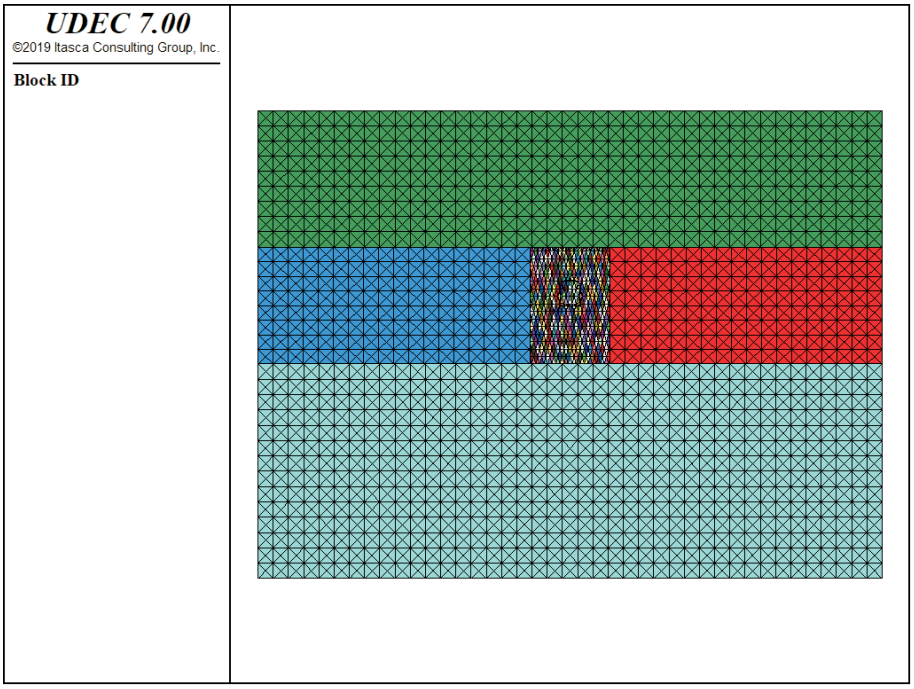

Figure 3.27 Full UDEC model of tunnel region

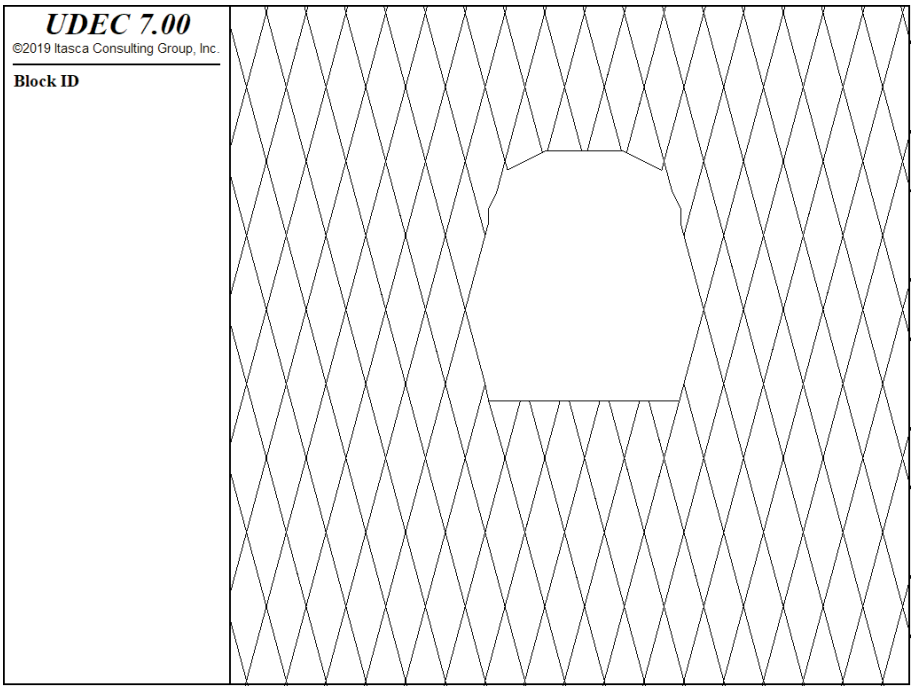

Figure 3.28 Close-up view of tunnel region

The model is zoned to provide a somewhat finer mesh in the vicinity of the excavation. The zone edge size is selected based on the characteristics of the dynamic analysis. Requirements for zoning in a dynamic simulation are described in Section 4.4 in **Special Features**. The zone generation commands are

```
gen edge 3.5  range  -10.33  10.33  15  45
gen quad 4.0
```

In this analysis, failure is assumed to be confined to the joints; blocks behave as elastic material. The commands to assign block properties are

```
group zone 'rock'
zone model elas density 2.34E3 bulk 8.39E9 shear 6.29E9 range group 'rock'
```

In this example, all blocks are assigned the group name **rock**.

The joints have a friction angle of 20°, cohesion of 1 MPa and zero tensile strength. The cohesion is eliminated if the joint fails in shear or tension (**block contact cmodel assign residual**). The commands are

```
bl cont group 'joint'
bl cont cmod assign resid st-s 1E10 st-n 1E10 fric 20 coh 1E6 ...
  range group 'joint'
; new contact default
bl cont cmod default residual st-s 1E10 st-n 1E10 fric 20 coh 1E6
```

All joints are assigned the group name **joint**, and any new contacts created during the simulation are assigned the same joint model and properties as these joints by using the **block contact cmodel default** command.

The initial vertical stress state is established by gravitational loading. The initial horizontal stress distribution is assumed equal to the vertical stress distribution. The boundary and initial conditions to bring the model to this initial equilibrium state are assigned via

```
bl grid apply vel-x 0 range -81,-79 -40,80
bl grid apply vel-x 0 range 79,81 -40,80
bl grid apply vel-y 0 range -80,80 -41,-39
block insitu stress -1872000.0,0.0,-1872000.0 grad-x 0.0,0.0,0.0 &
       grad-y 23400.0,0.0,23400.0
block mech gravity=0.0 -10.0
```

The initial stage is monitored and solved with the commands

```
bl grid history disp-y 0.0,35.0
block solve ratio 1.0E-7 elastic
```

Note that the **block solve elastic** command is used to minimize any effect of inertial loads during this initial equilibrium calculation.

Equilibrium is confirmed by plotting the unbalanced force and vertical displacement histories. The limiting equilibrium ratio in the **block solve** command is reduced from 10^{-5} to 10^{-7} to ensure that

equilibrium has been reached. Figure 3.29 shows that the vertical displacement at (x = 0, y = 35) has converged to a constant value.

The model state is saved at this stage:

```
save step1.sav
```

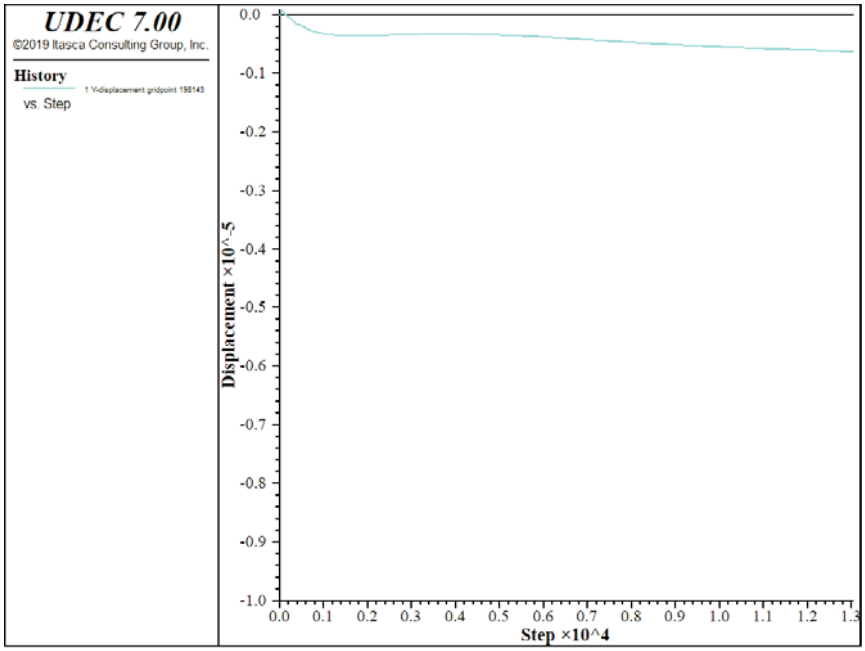

Figure 3.29 Vertical displacement history monitored at (x = 0, y = 35)

In the next stage, the tunnel is excavated instantaneously via the commands

```
block delete range pos-x -2.6,2.6   pos-y 30,35
block delete range pos-x -2.46,2.46 pos-y 34.6,36.5
block delete range pos-x -2.9,3.2   pos-y 31.9,32.6
```

This is the most common way to perform an excavation with *UDEC*, and it assumes that the excavation is made suddenly (e.g., by explosion). The resulting nonlinear response of the model will depend on the rate of unloading, so the modeler must decide whether this method of excavation in *UDEC* is appropriate to the physical problem. Alternatively, the excavation can be made by reducing the stresses along the excavation boundary gradually. A different response may result. The effect of path-dependent loading is discussed in more detail in Section 3.11.3.

Note that the **block delete** commands also delete blocks along the sidewalls of the tunnel. If these blocks are not deleted, they detach from the tunnel walls during this stage.

The solution for the second stage is found by monitoring displacement at locations around the tunnel:

```
block gridpoint hist dis-y 0,36.2
```

```
block gridpoint hist dis-x 2.5,32.5
```

The y-displacement is recorded at the crown of the tunnel, and the x-displacement at the wall midpoint. The solution is started by the command

```
block solve ratio 1.0E-6
```

A limiting equilibrium ratio of 10^{-6} is sufficient in this case to equilibrate the model. When the **block solver** calculation stops, the displacement histories are plotted. Figure 3.30 shows that the displacements have converged to a value of 0.03 mm (inward) at the crown and 0.09 mm (inward) at the springline. A plot of the displacement vectors and stress state around the tunnel at equilibrium is shown in Figure 3.31.

The model is now saved before beginning the dynamic loading stage:

```
model save 'step2.sav'
```

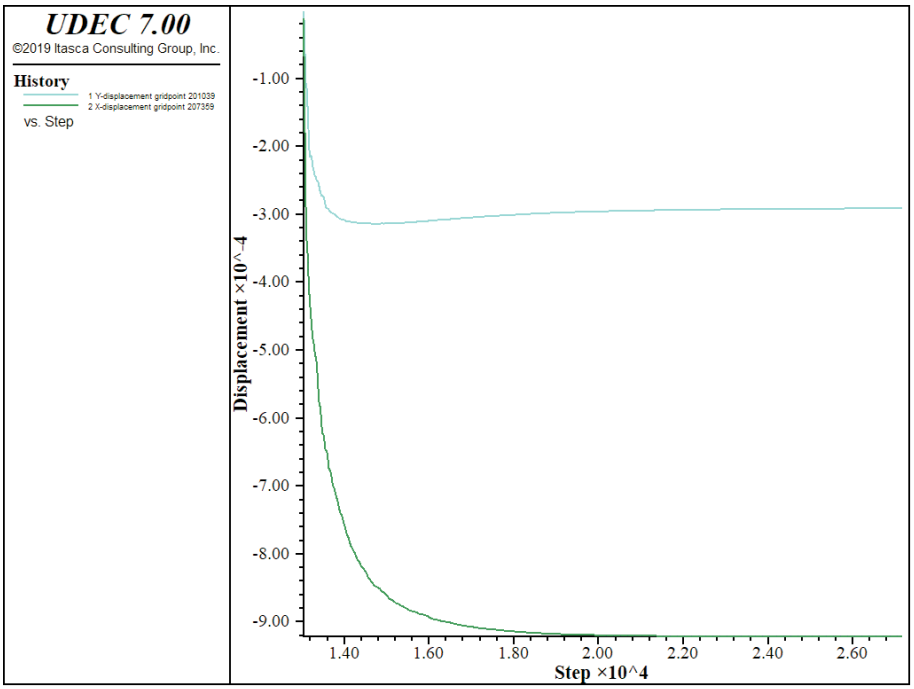

Figure 3.30 Displacement histories monitored at crown (hist 1) and at wall midpoint (hist 2) of the tunnel

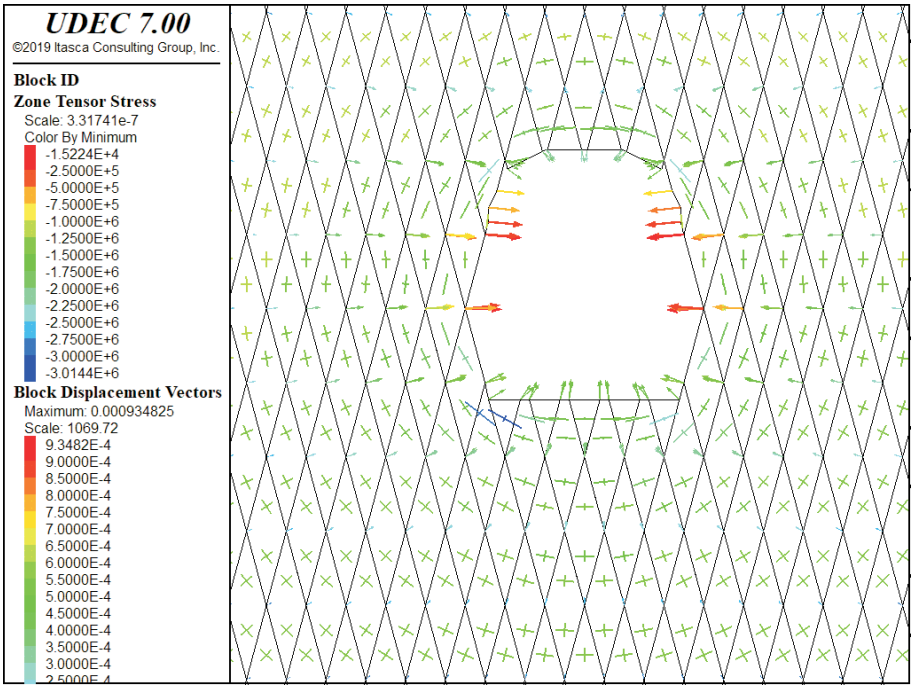

Figure 3.31 Stress-equilibrium after tunnel excavation

The dynamic wave is a simple representation of seismic loading; a 10 Hz sinusoidal shear wave is applied to the model base and allowed to propagate upward. The maximum velocity at the base of the model is approximately 1 m/sec. The commands for the dynamic analysis stage are listed in Example 3.22:

Example 3.22 Stability analysis of an underground excavation – seismic loading

```
model rest 'step2.sav'
; generate free-field
block edge apply dynamic-free-field
; apply dynamic boundary conditions
block gridpoint apply viscous-x bulk 8.39E9 shear 6.29e9 density 2.34e3 ...
  range position-x -80.1 80.1 position-y -40.1 -39.9
block edge apply property bulk 8.39E9 shear 6.29e9 density 2.34e3 ...
  range position-x -80.1 80.1 position-y -40.1 -39.9
block gridpoint apply viscous-y bulk 8.39E9 shear 6.29e9 density 2.34e3 ...
  range position-x -80.1 80.1 position-y -40.1 -39.9
block edge apply property bulk 8.39E9 shear 6.29e9 density 2.34e3 ...
  range position-x -80.1 80.1 position-y -40.1 -39.9
; amplitude of shear wave: 1 m/sec  freq = 10 Hz. duration 1 sec.
block edge apply stress 0.0 -7.68e6 0.0 history sine 10 1 ...
  range position-x -80.1 80.1 position-y -40.1 -39.9
block mechanical damping 0.0001 10 stiff
```

```
block gridpoint init velocity-x 0
block gridpoint init velocity-y 0
block gridpoint init displacement-x 0
block gridpoint init displacement-y 0
hist reset
block mechanical time 0
block gridpoint history velocity-y 0.0 36.2
block gridpoint history displacement-y 0.0 36.2
block gridpoint history velocity-x 2.5 32.5
block gridpoint history displacement-x 2.5 32.5
block gridpoint history velocity-x 0.0 -40.0
block gridpoint history velocity-x 0.0 80.0
block zone history stress-xy 0.0 -40.0
block mechanical history time-total
model save 'step4.sav'
;
; cycle and save 'in 0.2 sec increments
;
block contact tolerance overlap 0.5
block cycle time 0.2
model save 'step4a.sav'
;
block cycle time 0.2
model save 'step4b.sav'
;
```

These commands apply a sinusoidal shear wave and viscous boundary condition to the base of the model and free-field boundaries to the sides. See Section 4.5 in **Special Features** for an explanation of these boundary conditions. Note that the applied stress is double the actual value in order to compensate for the viscous boundary (see Section 4.7 in **Special Features**).

During the dynamic calculation stage, we switch to Rayleigh damping to simulate the actual material damping for this example. The command **DAMP 0.0001 10 stiff** applies stiffness-proportional damping at .01% of 10 Hz frequency. This is a very low damping level that is specifically intended to damp high frequency components during the transmission of the wave. (See Section 4.3.3 in **Special Features** for guidelines on selecting damping parameters for dynamic analyses.)

The sinusoidal shear wave is applied for a time period of 1.0 s. The calculation is performed in 0.2 s increments for a total duration of 0.4 s. Figure 3.32 displays the tunnel at 0.2 s after application of the seismic loading, and Figure 3.33 displays the tunnel at 0.4 s. Heave occurs at the invert, and roof blocks are shown to be unstable.

The wave reflected from the ground surface is responsible for the roof blocks becoming unstable. As shown in Figure 3.34, the displacement magnitude at the tunnel wall midpoint (history 4) is nearly doubled at 0.25 s after application of the seismic load when the reflected wave reaches the excavation.

The modeling sequence can now be repeated (e.g., **RESTORE step2.sav**) with cable elements or structural beam elements installed in order to investigate the effect of support on stability. If different material properties for the blocks or joints are used, the static calculation for the tunnel must be solved again (**RESTORE step1.sav**). If a different tunnel location or orientation of joint sets is evaluated, it will be necessary to regenerate the model and bring it to an initial equilibrium state again. Always remember that the model *must* be at an equilibrium state when the loading change is made.

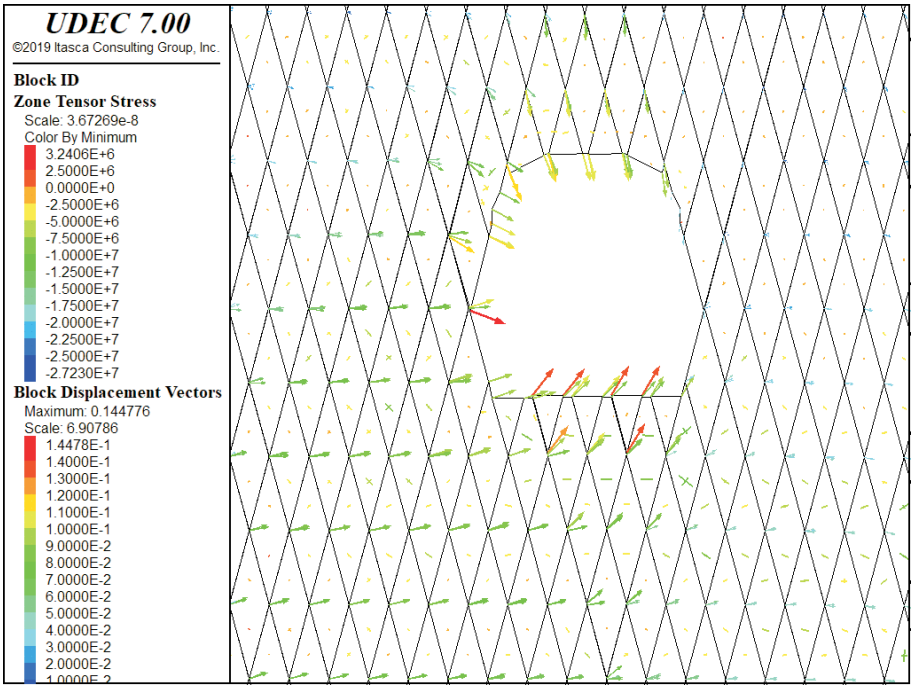

Figure 3.32 Unstable blocks identified at 0.2 s of seismic loading

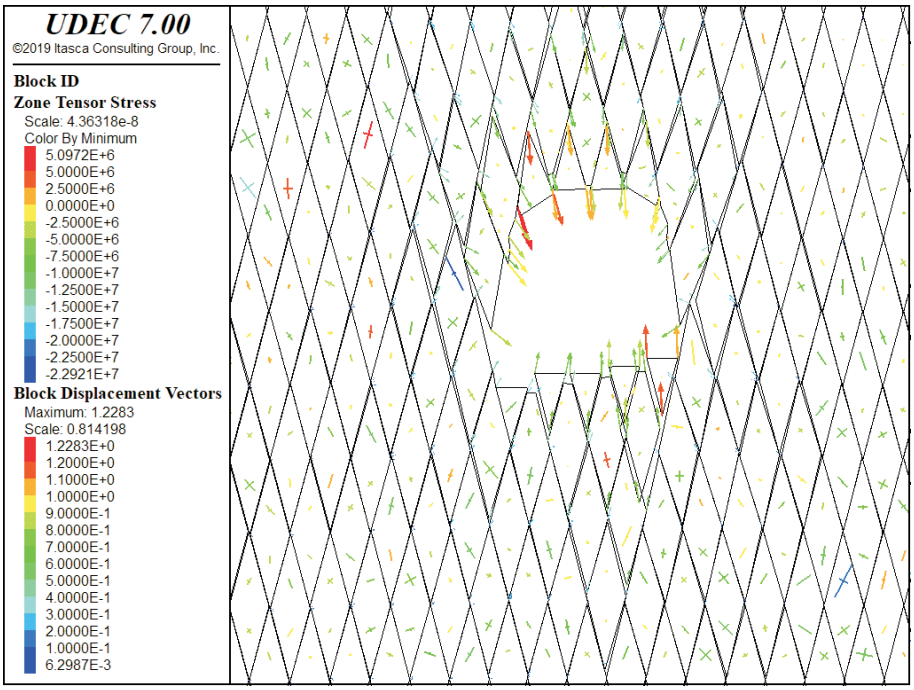

Figure 3.33 Unstable blocks identified at 0.4 s of seismic loading

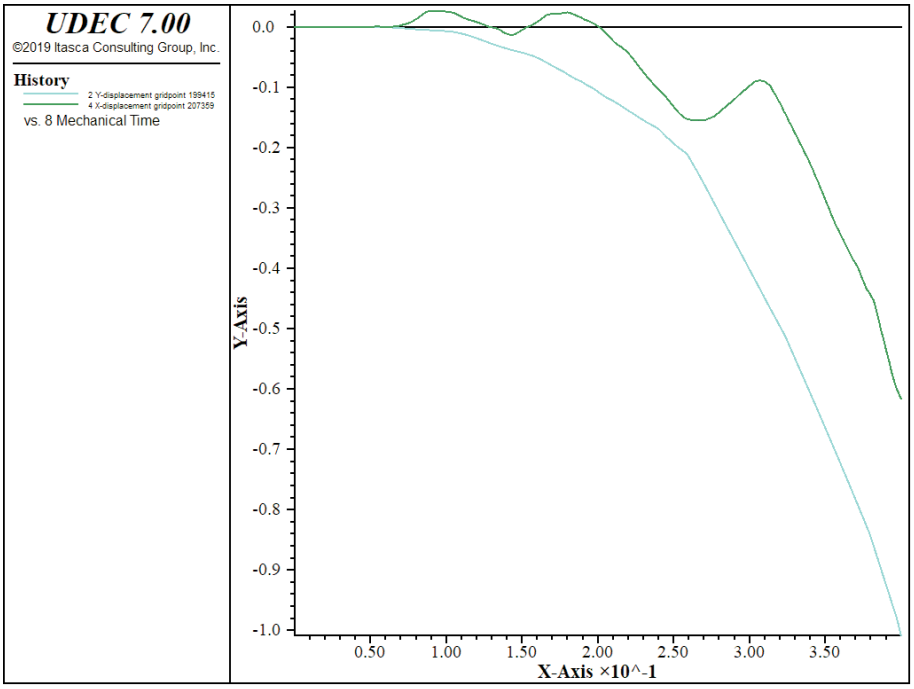

Figure 3.34 Displacement histories at crown (hist 2) and at wall midpoint (hist 4) for a period of 0.4 s of seismic loading

3.7 Choice of Constitutive Model

This section provides an overview of the deformable block and joint constitutive models in *UDEC*, as well as recommendations for when to use a model. Section 1 in **Constitutive Models** presents background information on the block constitutive model formulations. The joint models are described in Section 1.2.4.4 in **Theory and Background** and Sections 2.2 and 3 in **Constitutive Models**.

3.7.1 Deformable-Block Material Models

There are fourteen built-in block material models in *UDEC**:

Section 4 in **Constitutive Models**).:

(1) null (**block zone cmodel assign null**);

(2) elastic, isotropic (**block zone cmodel assign elastic**);

(3) elastic, anisotropic (**block zone cmodel assign anisotropic**);

(4) Drucker-Prager plasticity (**block zone cmodel assign drucker-p**);

(5) Mohr-Coulomb plasticity (**block zone cmodel assign mohr-c**);

(6) ubiquitous-joint (**block zone cmodel assign ubiquitous-joint**);

(7) strain-hardening/softening (**block zone cmodel assign strain-softening**);

(8) bilinear strain-hardening/softening ubiquitous joint (**block zone cmodel assign softening-ubiquitous**);

(9) modified Cam-clay (**block zone cmodel assign modified-cam-clay**);

(10) Hoek-Brown (**block zone cmodel assign hoek-brown**);

(11) Hoek-Brown-PAC (**block zone cmodel assign hoek-brown-pac**);

(12) Cap-yield (**block zone cmodel assign cap-yield**);

(13) simplified Cysoil (**block zone cmodel assign cap-yield-simplified**); and

(14) double-yield (**block zone cmodel assign double-yield**).

---

* There are also several optional models that simulate viscoelastic and viscoplastic (creep) behavior available (see Section 1 in **Creep Material Models**). In addition, users can modify the built-in models or create their own constitutive models by using a user-written dynamic link library (DLL) written in C++ (see writing new constitutive models in help in *UDEC*).

A model can be assigned to an entire block or zones within a portion of a block or group of blocks using the **block zone cmodel assign** command. Properties are also assigned directly to zones using the **block zone property** command.*

Each model is developed to represent a specific type of constitutive behavior commonly associated with geologic materials. The *null* model is used to represent material that is removed from the model. The *elastic, isotropic* model is valid for homogeneous, isotropic, continuous materials that exhibit linear stress-strain behavior. The *elastic, transversely isotropic* model is appropriate for elastic materials that exhibit a well-defined elastic anisotropy. The *Drucker-Prager plasticity* model is a simple failure criterion in which the shear yield stress is a function of isotropic stress. The *Mohr-Coulomb plasticity* model is used for materials that yield when subjected to shear loading, but the yield stress depends on the major and minor principal stresses only; the intermediate principal stress has no effect on yield. The *ubiquitous-joint* model corresponds to a Mohr-Coulomb material that exhibits a well-defined strength anisotropy due to embedded planes of weakness. The *strain-softening* model is based upon the Mohr-Coulomb model, but is appropriate for materials that show a degradation or increase in shear strength when loaded beyond the initial failure limit. The *bilinear strain-hardening/softening ubiquitous-joint* model is a generalization of the ubiquitous-joint model that allows the strength properties for the matrix and the joint to harden or soften. The *double-yield* model is an extension of the strain-softening model to simulate irreversible compaction as well as shear yielding. The *modified Cam-clay* model accounts for the influence of volume change on deformability and on resistance to failure. The *Hoek-Brown-PAC* model is an empirical relation that is a nonlinear failure surface representing the strength limit for isotropic intact rock and rock masses. This model also includes a plasticity flow rule that varies as a function of the confining stress level. The *Hoek-Brown* model is a variation of the Hoek-Brown model that characterizes post-failure plastic flow by dimple flow rule choices based upon dilation angle. The *cap-yield* model is a simple strain-hardening/softening model for soils, characterized by a frictional Mohr-Coulomb failure envelope and nonlinear volumetric cap. The *cap-yield-simplfied* model is a simplification of the *cap-yield* model without the volumetric cap.

The material models in *UDEC* are primarily intended for applications related to geotechnical engineering (e.g., underground construction, mining, slope stability, foundations, and earth and rockfill dams). The are two considerations when selecting a constitutive model for a particular engineering analysis:

1. What are the known characteristics of the material being modeled?
2. What is the intended application of the model analysis?

---

* Please note that the **block change model** command is also available to assign constitutive models to blocks. However, this command only permits models to be assigned to entire blocks, and is limited to only a few models.

Table 3.2 presents a summary of the *UDEC* block models, along with examples of representative materials and possible applications of the models. The elastic block model is generally applicable for cases in which slip along discontinuities is the predominant mechanism for failure. The Mohr-Coulomb model should be used when stress levels are such that failure of intact material is expected. Mohr-Coulomb parameters for cohesion and friction angle are usually available more often than other properties for geo-engineering materials.

Table 3.2 UDEC constitutive models

Model	Representative Material	Example Application
null	void	holes, excavations, regions in which material will be added at later stage
elastic	homogeneous, isotropic continuum; linear stress-strain behavior	manufactured materials (e.g., steel) loaded below strength limit; factor-of-safety calculation
transversely isotropic elastic	thinly laminated material exhibiting elastic anisotropy (e.g., slate)	laminated materials loaded below strength limit
Drucker-Prager plasticity	limited application; soft clays with low friction	common model for comparison to implicit finite-element programs
Mohr-Coulomb plasticity	loose and cemented granular materials; soils, rock, concrete	general soil or rock mechanics (e.g., slope stability and underground excavation)
strain-hardening / softening Mohr-Coulomb	granular materials that exhibit nonlinear material hardening or softening	studies in post-failure (e.g., progressive collapse, yielding pillar, caving)
ubiquitous-joint	thinly laminated material exhibiting strength anisotropy (e.g., slate)	excavation in closely bedded strata
bilinear strain-hardening/softening ubiquitous-joint	laminated materials that exhibit nonlinear material hardening or softening	studies in post-failure of laminated materials
double-yield	lightly cemented granular material in which pressure causes permanent volume decrease	hydraulically placed backfill
modified Cam-clay	materials for which deformability and shear strength are a function of volume change	geotechnical construction on clay
Hoek-Brown-PAC plasticity	isotropic rock material	geotechnical construction in rock
Hoek-Brown plasticity	isotropic rock material	geotechnical construction in rock including factor-of-safety calculation
cap-yield	soils that exhibit decreasing stiffness as plastic strains develop	geotechnical construction in soft soils
cap-yield-simplified	simplified version of Cysoil model to simulate hyperbolic strain-strain behavior	geotechnical construction in soft soils

The ubiquitous-joint, strain-softening and double-yield plasticity models are actually variations of the Mohr-Coulomb model. These models will produce results identical to those for Mohr-Coulomb if the additional material parameters are set to high values. The Drucker-Prager model is a simpler failure criterion than Mohr-Coulomb, but it is not generally suitable for representing failure of geologic materials. It is provided mainly to allow comparison of *UDEC* to other numerical programs that have the Drucker-Prager model but not the Mohr-Coulomb model. Note that at zero friction, the Mohr-Coulomb model degenerates to the Tresca model, while the Drucker-Prager model degenerates to the von Mises model.

The Drucker-Prager and Mohr-Coulomb models are the most computationally efficient plasticity models; the other plasticity models require increased memory and additional time for calculation. For example, plastic strain is not calculated directly in the Mohr-Coulomb model.

(see Section 1 in **Constitutive Models**). If plastic strain is required, the strain-softening or double-yield model must be used. These two models are primarily intended for applications in which the post-failure response is important (e.g., yielding pillars, caving or backfilling studies).

The tensile failure criterion is identical in the Mohr-Coulomb, ubiquitous-joint, strain-softening and double-yield plasticity models. The tensile failure criterion is defined separately from the shear and volumetric strength criteria. Tensile failure is defined by a tensile strength limit, and post-failure is governed by an associated plasticity flow rule. For the Mohr-Coulomb and ubiquitous-joint models, the value assigned to the tensile strength is set to zero (instantaneous softening) when tensile failure occurs and flag-brittle is set true. Tensile softening can be controlled with the strain-hardening/softening or double-yield model. By default, the tensile strength does not change upon failure in these models (flag-brittle = false).

The double-yield and modified Cam-clay models both take into account the influence of volumetric change on material deformability and failure characteristics. In both models, tangential bulk and shear moduli are functions of plastic volumetric deformation.

The differences between the two models are summarized as follows.

In the Cam-clay model:

1. The elastic deformation is nonlinear, with the elastic moduli depending on mean stress.
2. Shear failure is affected by the occurrence of plastic volumetric deformation; the material can harden or soften, depending on the degree of preconsolidation.
3. As shear loading increases, the material evolves toward a critical state at which unlimited shear strain occurs with no accompanying change in specific volume or stress.
4. There is no resistance to tensile mean stress.

In the double-yield model:

1. Elastic moduli remain constant during elastic loading and unloading.

2. Shear and tensile failure are not coupled to plastic volumetric change due to volumetric yielding. The shear yield function corresponds to the Mohr-Coulomb criterion, and the tensile yield is evaluated based on a tensile strength.

3. Material hardening or softening, upon shear or tensile failure, is defined by tables relating friction angle and cohesion to plastic shear strain, and tensile strength to plastic tensile strain.

4. The cap pressure is not influenced by the amount of shear or tensile plastic deformation.

5. A tensile strength limit and tensile softening can be defined.

The double-yield model was initially developed to represent the behavior of mine backfill material, for which preconsolidation pressures are low. The modified Cam-clay model is more applicable to soils such as soft clays for which preconsolidation pressures can have a significant effect on material behavior.

The Hoek-Brown model combines the generalized Hoek-Brown criterion with a plasticity flow rule that varies as a function of the confining stress level. At low confining stress, the volumetric expansion at yield is high, associated with axial splitting and wedging effects. At high confining stress, the material approaches a non-dilatant condition.

The modified Hoek-Brown model characterizes post-failure plastic flow by simple flow rule choices, and also contains a tensile strength limit similar to that used by the Mohr-Coulomb model. A factor-of-safety calculation based upon the shear-strength reduction method can be performed with this version of the Hoek-Brown model.

The cap-yield model is a modification of the double-yield model with a nonlinear volumetric cap. Cap hardening, friction hardening and compaction/dilation laws can be specified in the cap-yield model to customize the model to fit different characteristics of soft soil behavior.

A simplified version of the cap-yield model (the cap-yield-simplified model) provides built-in friction-hardening laws and hyperbolic stress-strain parameters as direct input.

3.7.2 Joint Material Models

There are four built-in models and one optional model available to represent the material behavior of discontinuities*:

(1) point contact – Coulomb slip (**block contact cmodel assign point**);

(2) joint area contact – Coulomb slip (**block contact cmodel assign area**);

(3) joint area contact – Coulomb slip with residual strength (**block contact cmodel assign residual**);

(4) continuously yielding (**block contact cmodel assign cy**); and

(5) Barton-Bandis joint (**blo cont cmod assign barton-bandis**), optional model **.

The joint model can be assigned directly to individual, or groups of, contacts using the **block contact cmodel assign** command. Properties are also assigned directly to the contacts using the **block contact property** command. If new contacts are created during a *UDEC* simulation, these contacts can be assigned a joint model by using the **block contact cmodel default** command.***

The joint constitutive models are designed to be representative of the physical response of rock joints. The *point contact* model represents the contact between two blocks where the contact area is very small relative to the dimensions of the block. The *area contact* model is intended for closely packed blocks with area contact. The model provides a linear representation of joint stiffness and yield limit, and is based upon elastic stiffness, frictional, cohesive and tensile strength properties, and dilation characteristics common to rock joints. The *residual-strength* version of this model simulates displacement-weakening of the joint by loss of frictional, cohesive and/or tensile strength at the onset of shear or tensile failure. The *continuously yielding* joint model is a more complex model, which simulates continuous weakening behavior as a function of accumulated plastic-shear displacement. The *Barton-Bandis* model is also a nonlinear joint model that directly utilizes index properties from laboratory tests on joints derived by Drs. Nick Barton and Stavros Bandis at the Norwegian Geotechnical Institute (e.g., Bandis et al. 1983).

---

* Users can modify the built-in joint models or create their own joint constitutive models by using a user-written dynamic link library (DLL) written in C++ (see writing new constitutive models in Section 4 in **Constitutive Models**).

** The development of the Barton-Bandis joint model was sponsored by the Norwegian Geotechnical Institute, NGI; the model is available at an additional cost.

*** The joint models can also be assigned to one or more contacts by using the **bloc contact change model** command. Joint model properties are then specified with the **block contact property material n** command for material property numbers. With this approach, joint properties are not assigned directly to contacts, but to material numbers, and the numbers are assigned to the contacts with the **block contact change material** command.

Table 3.3 summarizes the *UDEC* joint models and presents examples of representative materials and possible applications. The area contact Coulomb slip model is most applicable for general engineering studies. Coulomb friction and cohesion properties are usually available more often than other joint properties. If it is anticipated that both area contact and point contact between blocks will occur within a model, it is acceptable to specify only the area contact model. Point contacts will be assigned automatically when the minimum contact area is equal to twice the model rounding length. The point contact model should be specified if point contacts are predominant (e.g., in a loosely packed system of irregularly shaped particles).

Table 3.3 UDEC joint constitutive models

Model	Representative Material	Example Application
point contact	limited application; particulate material; loosely compacted blocks of irregular shape	slope stability in highly disturbed, broken and fractured rock
area contact	joints, faults, bedding planes in rock	general rock mechanics (e.g., underground excavation)
area contact with displacement weakening	rock joints displaying distinct peak/residual strengths	general rock mechanics
continuously yielding	rock joints displaying progressive damage and hysteretic behavior	cyclic loading and load reversal with predominant hysteretic loop; dynamic analysis
Barton-Bandis	rock joints defined by Barton-Bandis index properties	estimation of changes in hydraulic apertures

Both the continuously yielding and Barton-Bandis joint models are empirical expressions that require more detailed knowledge of the joint behavior. The properties for the continuously yielding model are derived from laboratory test results relating joint shear stress to shear and normal displacement. The Barton-Bandis joint properties are determined from joint index testing, as previously mentioned. It is always recommended that initial studies be based upon the Coulomb slip model first, in order to develop a fundamental understanding of joint response before applying the more complex joint models. This is discussed in more detail in the following section. Example comparisons of the Coulomb slip model and the continuously yielding model are shown in Section 1 in the **Verification Problems**. A demonstration of the response of the continuously yielding model and the required properties are provided in Section 2 in **Constitutive Models**.

3.7.3 Selection of an Appropriate Model

A problem analysis should always start with simple block and joint material models; in most cases, an elastic block model (**block zone cmodel assign elastic**) and a joint area contact Coulomb slip model (**block contact cmodel assign area**) should be used first. The elastic block model only requires three material parameters: mass density, bulk modulus and shear modulus (see Section 3.8.1.2). The Coulomb slip model requires six parameters: normal and shear stiffness, friction angle, cohesion, tensile strength and dilation angle. Estimates and references for these properties are given in Section 3.8.2. These material models provide a simple perspective of stress-deformation behavior in the *UDEC* model; the results of these analyses can help the user assess whether a more complex (or simpler) material model is needed to describe the block or joint behavior. For example, if the stresses and deformations in the blocks are low compared to the joint movements, then a simpler, rigid block model may be sufficient.

It is often helpful to run simple tests of the selected material model before using it to solve the full-scale, boundary-value problem. This can provide insight into the expected response of the model compared to the known response of the physical material.

The following example illustrates the use of a simple test model. The problem application is the analysis of joint slip around an underground excavation. A simple model is created to evaluate the adequacy of the Coulomb slip model and the continuously yielding model to represent the response of a joint subjected to shear loading. The test is a simulation of a direct shear test, which consists of a single horizontal joint that is first subjected to a normal confining stress and then to a unidirectional shear displacement. Figure 3.35 shows the model; the joint is defined by four contacts.

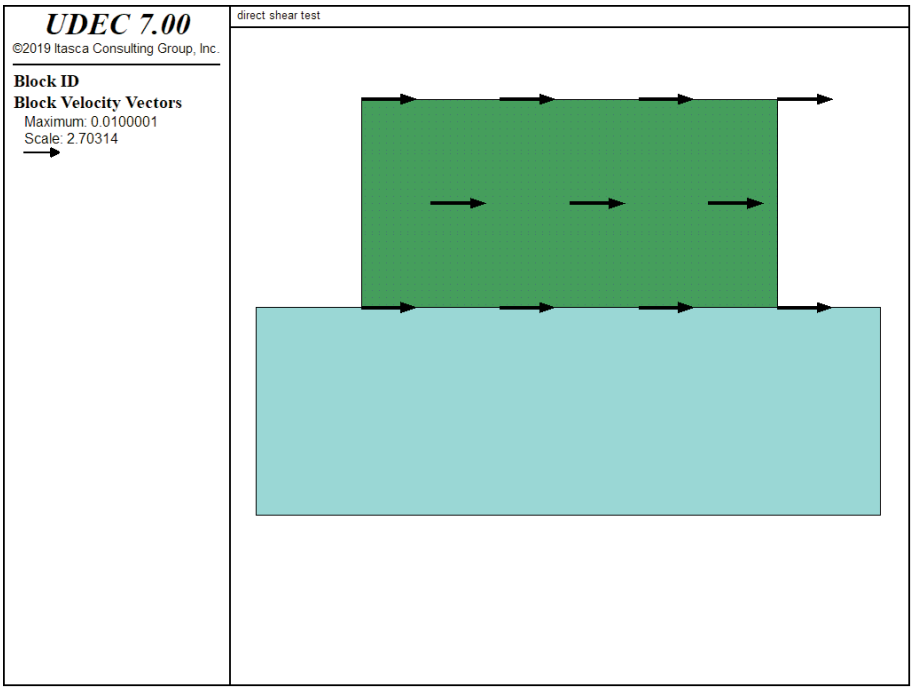

Figure 3.35 Direct shear test model

First, a normal stress of 10 MPa, which is representative of the confining stress acting on the joint, is applied. A horizontal velocity is then applied to the top block to produce a shear displacement that is also representative of the displacements expected in the problem application. For demonstration purposes, we only apply a small shear displacement (of less than 1 mm) to this model.

The average normal and shear stresses and normal and shear displacements along the joint are measured with a *FISH* function (**av_str**). With this information, we can determine the peak and residual shear strengths and dilation that are produced with the different models. The data file for this test using the Coulomb slip model is contained in Example 3.23:

Example 3.23 Direct shear test with Coulomb slip model

```
model new
; create shear test model
model title 'direct shear test'
block tolerance corner-round-length 0.001
block tolerance minimum-edge-length 0.002
block create polygon -5E-2 -0.1 -5E-2 0.1 0.25 0.1 0.25 -0.1
block cut crack -1 0 1 0
block cut crack 0 0.1 0 0
block cut crack 0.2 0.1 0.2 0
block delete range pos-x -5E-2 0 pos-y 0 0.1
block delete range pos-x 0.2 0.25 pos-y 0 0.1
block zone gen quad 0.4 0.11 range pos-x 0 1 pos-y -1 0
block zone gen quad 0.07 0.11 range pos-x 0 1 pos-y 0 1
block zone group 'block'
block zone cmodel assign elastic density 2.6E-3 bulk 4.5E4 shear 3E4 ...
  range group 'block'
;
; Coulomb joint model
block contact group 'joint'
block contact cmodel assign area stiffness-shear 4E4 ...
  stiffness-normal 4E4 friction 30 dilation 6 dilation-limit 4E-4 ...
  range group 'joint'
; new contact default
block contact cmodel default area stiffness-shear 4E4 ...
  stiffness-normal 4E4 friction 30 dilation 6 dilation-limit 4E-4
;  set add_dil on
;
; Apply boundary conditions
block gridpoint apply velocity-x 0 range pos-x -6E-2 -4E-2 pos-y -1 1
block gridpoint apply velocity-x 0 range pos-x 0.24 0.26 pos-y -1 1
block gridpoint apply velocity-y 0 range pos-x -1 1 pos-y -0.11 -9E-2
block edge apply stress 0.0 0.0 -10.0 range pos-x -1 1 pos-y 9E-2 0.11
block solve ratio 1.0E-5
model save 'direct1.sav'
```

```
;
; functions to calculate average joint stresses
; and average joint displacements
;
;Name:ini_jdisp
fish define ini_jdisp
   njdisp0 = 0.0
   sjdisp0 = 0.0
   ic = block.contact.head
   loop while ic # 0
      njdisp0 = njdisp0 + block.contact.disp.normal(ic)
      sjdisp0 = sjdisp0 + block.contact.disp.shear(ic)
      ic = block.contact.next(ic)
   endloop
end
@ini_jdisp
;
;Name:av_str
fish define sstav
   whilestepping
   sstav1 = 0.0
   nstav = 0.0
   njdisp = 0.0
   sjdisp = 0.0
   ncon = 0
   jl   = 0.2              ; joint length
   ic = block.contact.head
   loop while ic # 0
      ncon = ncon+1
      sstav1 = sstav1 + block.contact.force.shear(ic)
      nstav = nstav + block.contact.force.normal(ic)
      njdisp = njdisp + block.contact.disp.normal(ic)
      sjdisp = sjdisp + block.contact.disp.shear(ic)
      ic = block.contact.next(ic)
   endloop
   if ncon # 0
      sstav1 = sstav1 / jl
      nstav = nstav / jl
      njdisp = (njdisp-njdisp0) / ncon
      sjdisp = (sjdisp-sjdisp0) / ncon
   endif
   sstav = sstav1
end
@sstav
;
block contact reset displacement
```

```
hist reset
fish history @sstav
fish history @nstav
fish history @njdisp
fish history @sjdisp
history interval 1
;
; Apply shear load by imposing x-velocity on top block
block gridpoint apply velocity-x 1E-2 ...
  range pos-x -1E-2 0.21 pos-y -1E-2 0.1
model save 'direct2.sav'
;
;  Change to displacement-weakening model
; block contact group 'residual joint'
; block contact cmodel assign residual stiffness-shear 4E4 ...
;   stiffness-normal 4E4 friction 35 cohesion 2 dilation 6 ...
;   dilation-limit 4E-4 cohesion-residual 0 friction-residual 30 ...
;   range group 'residual joint'
; new contact default
;block contact cmodel default residual stiffness-shear 4E4 ...
;   stiffness-normal 4E4 friction 35 cohesion 2 dilation 6 ...
;   dilation-limit 4E-4 cohesion-residual 0 friction-residual 30
;
;
block cycle 11500
model save 'direct3.sav'
return
```

The average shear stress versus shear displacement along the joint is plotted in Figure 3.36, and the average normal displacement versus shear displacement is plotted in Figure 3.37. These plots indicate that joint slip occurs for the prescribed model properties and conditions. The loading slope in Figure 3.36 is linear until a peak shear strength of approximately 6 MPa is reached. As indicated in Figure 3.37, the joint begins to dilate when the joint fails in shear, at roughly 0.15 mm shear displacement. Dilation occurs until the limiting shear displacement (**dilation-limit** = 0.4 mm) is reached for zero dilation. The maximum average dilation is approximately 0.027 mm.

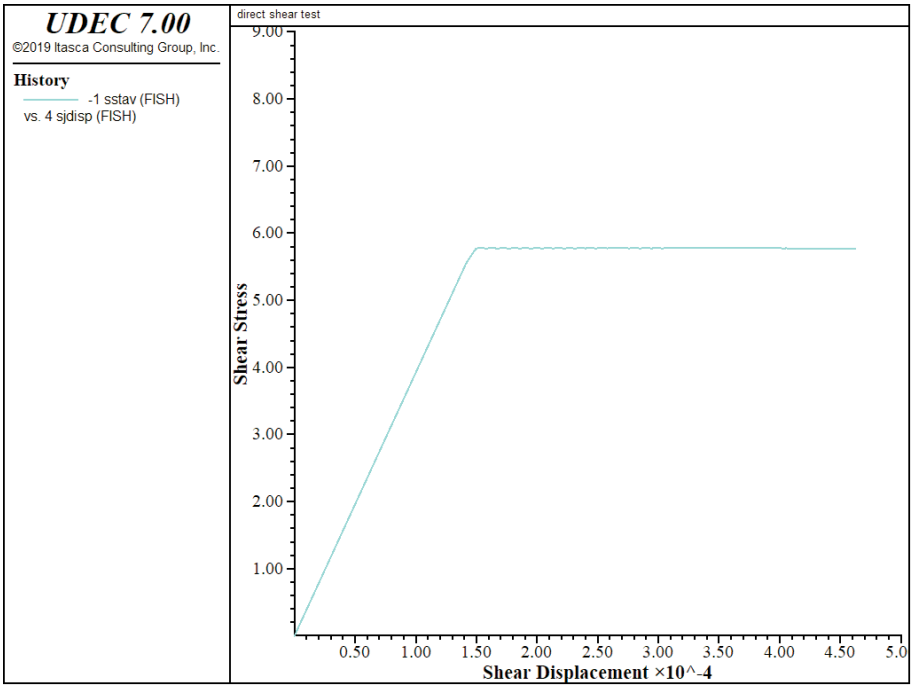

Figure 3.36 Average shear stress versus shear displacement – Coulomb slip model

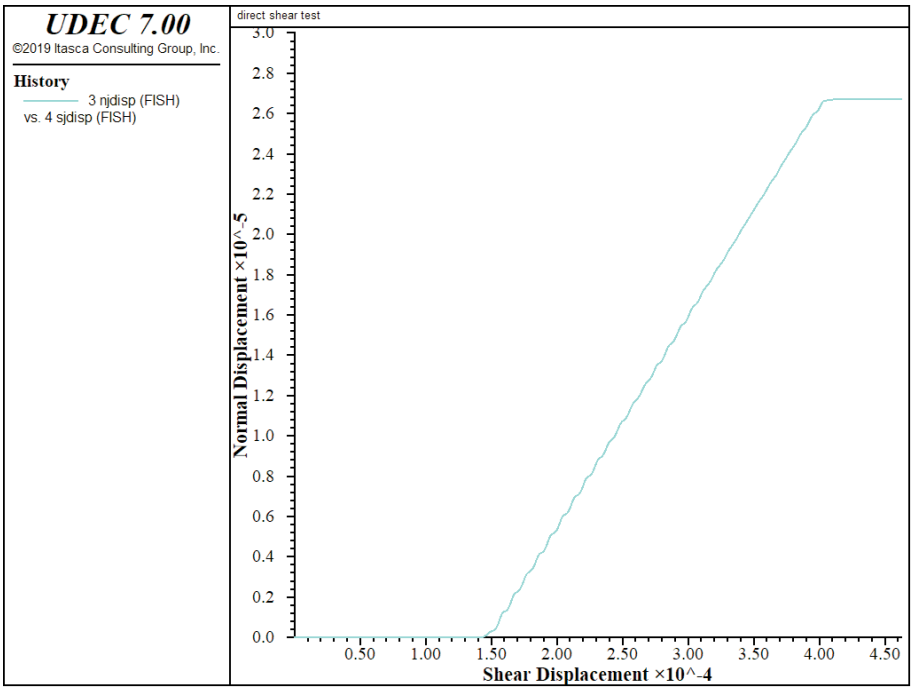

Figure 3.37 Average normal displacement versus shear displacement – Coulomb slip model

As these results indicate, the Coulomb slip model (**block contact cmodel assign area**) *only* defines a limiting shear strength value for the joint. The dilation that occurs after the joint begins to slip is approximated as a linear function of the dilation angle, with a dilation limit that is a function of the shear displacement. (These functions are described in Section 1.2.4 in **Theory and Background**.)

Other modifications to the joint behavior are also available for the Coulomb slip model. For example, a displacement-weakening behavior can be approximated by including the dilation in the effective friction angle when the joint is slipping. This behavior is invoked by using the command **block contact add-dilation on** with the **block contact cmodel assign area** model. (See Example 3.23.) The effective friction is obtained by adding the dilation angle to the input friction angle. The residual strength occurs when the limiting shear displacement for zero dilation is reached. The response with this option is shown in Figures 3.38 and 3.39. Note that only the friction is adjusted in this model.

There are other ways the Coulomb model can be modified to simulate a peak and residual shear strength of the joint. One way is to use the **block contact cmodel assign residual** version of the Coulomb model. In Example 3.23, we add the lines (i.e., remove the comments)

```
group joint 'residual joint'
block contact cmodel assign residual st-n 4E4 st-s 4E4 friction 35 ...
  cohesion 2 dilation 6 dilation-limit 4E-4 coh-res 0 fric-res 30 ...
  range group 'residual joint'
; new contact default
block contact cmodel default residual st-s=4E4 st-n=4E4 friction=35 ...
  cohesion=2 dilation=6 dilation-limit=4E-4 coh-res=0 fric-res=30
```

before conducting the shear test on the joint. (Note that we still use the **block contact cmodel assign area** command for the initial normal loading phase. This prevents any loss in joint strength from occurring during the initial equilibrium calculation.) The results, shown in Figure 3.40, illustrate the peak and residual strengths that develop when using this model. For this model, friction angle, cohesion and tensile strength can be assigned peak and residual values; in the example, we reduce friction angle and cohesion. Note that the drop in strength occurs abruptly. The maximum dilation, as shown in Figure 3.41, is lower with the **block contact change model = 5** model for the same limiting shear displacement because the joint initially fails after more shear displacement has occurred (compare Figure 3.41 to Figure 3.37).

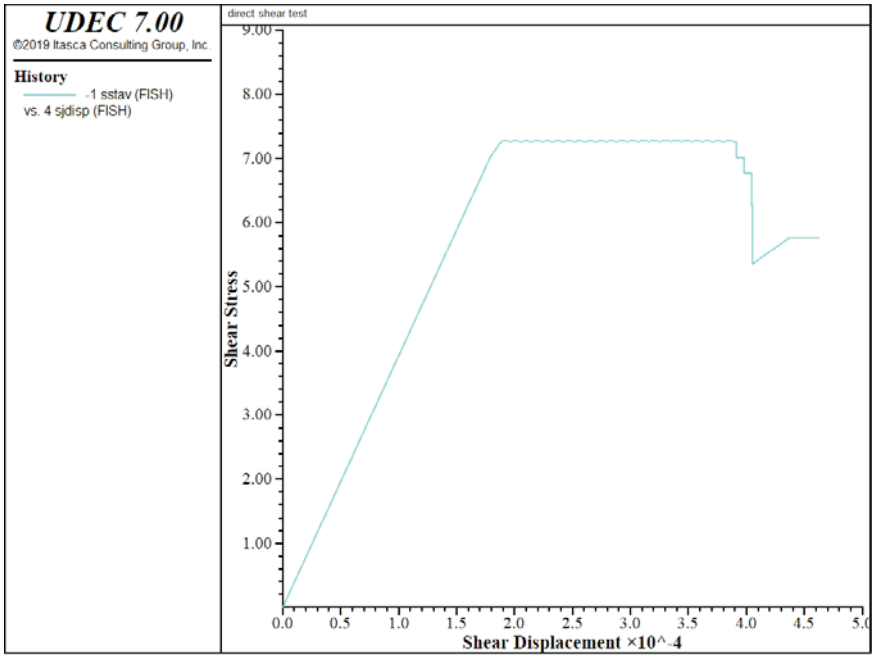

Figure 3.38 Average shear stress versus shear displacement – Coulomb slip model with dilation included in the effective friction angle

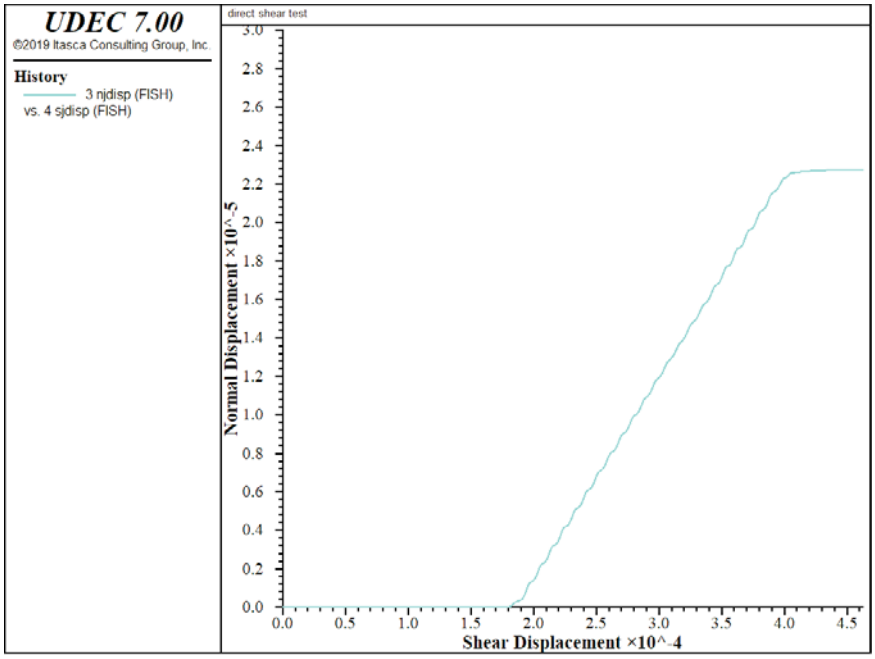

Figure 3.39 Average normal displacement versus shear displacement – Coulomb slip model with dilation included in the effective friction angle

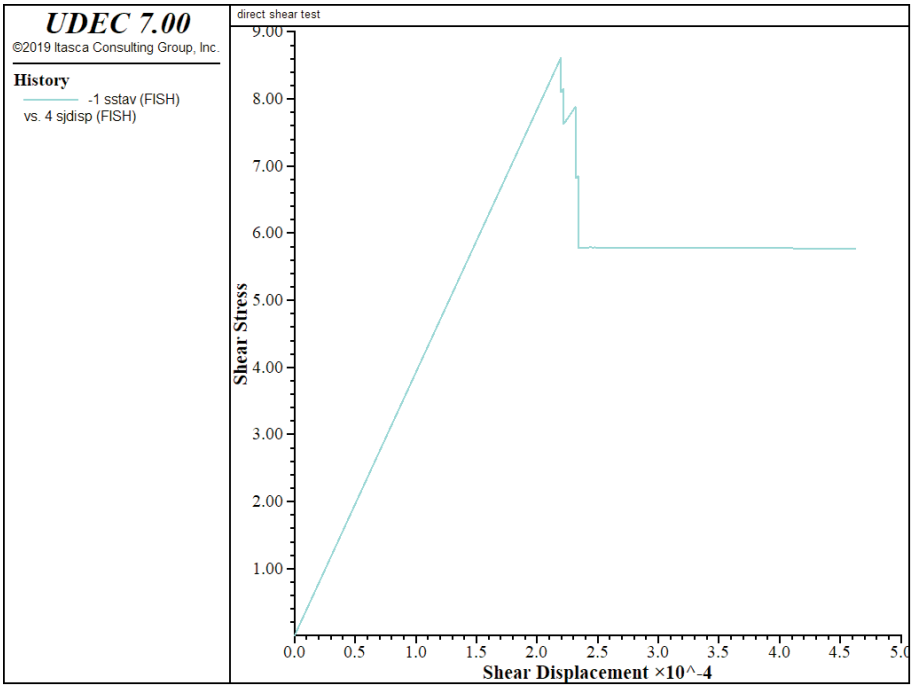

Figure 3.40 Average shear stress versus shear displacement – Coulomb slip model with peak and residual strength

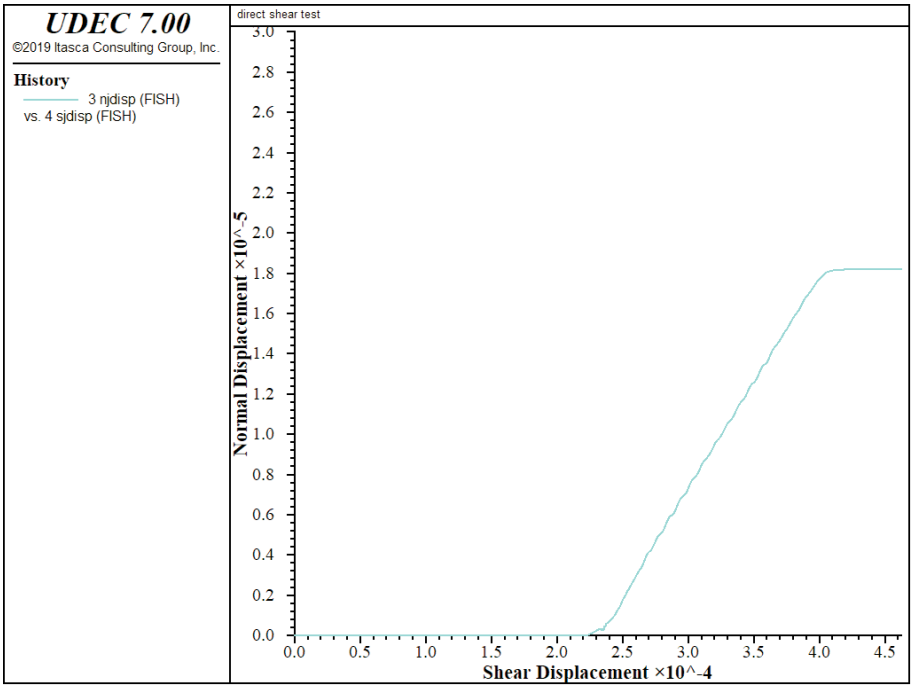

Figure 3.41 Average normal displacement versus shear displacement – Coulomb slip model with peak and residual strength

Alternatively, a user-defined function can be applied to reduce the joint strength. We can use *FISH* to access the joint properties (e.g., joint friction angle and cohesion). In Example 3.24, we reduce these properties linearly as a function of shear displacement. Note that the **block contact cmodel assign area** command is used to assign joint properties locally to the contacts. Then the joint model and properties can be accessed for each contact via the **block.contact.prop(ic,'cohesion')** *FISH* variable, where ***ic*** is the contact index number. (See the FISH functions in the Help in *UDEC*. The shear displacement at a contact is accessed using block.contact.disp.shear(ic)

The results of this test are shown in Figures 3.42 and 3.43.

Example 3.24 Direct shear test with joint properties controlled by a FISH function

```
model new
; create shear test model
model title 'direct Shear Test'
block tolerance corner-round-length 0.001
block tolerance minimum-edge-length 0.002
block create polygon -5E-2 -0.1 -5E-2 0.1 0.25 0.1 0.25 -0.1
block cut crack -1 0 1 0
block cut crack 0 0.1 0 0
block cut crack 0.2 0.1 0.2 0
block delete range pos-x -5E-2 0 pos-y 0 0.1
block delete range pos-x 0.2 0.25 pos-y 0 0.1
block zone gen quad 0.4 0.11 range pos-x 0 1 pos-y -1 0
block zone gen quad 0.07 0.11 range pos-x 0 1 pos-y 0 1
block zone group 'block'
block zone cmodel assign elastic density 2.6E-3 bulk 4.5E4 shear 3E4 ...
  range group 'block'
;
; Coulomb joint model
block contact group 'joint'
block contact cmodel assign area stiffness-shear 4E4 ...
  stiffness-normal 4E4 friction 35 cohesion 2 dilation 6 ...
  dilation-limit 4E-4 range group 'joint'
; new contact default
block contact cmodel default area stiffness-shear 4E4 ...
  stiffness-normal 4E4 friction 30 cohesion 2 dilation 6 ...
  dilation-limit 4E-4
;
; Apply boundary conditions
block gridpoint apply velocity-x 0 range pos-x -6E-2 -4E-2 pos-y -1 1
block gridpoint apply velocity-x 0 range pos-x 0.24 0.26 pos-y -1 1
block gridpoint apply velocity-y 0 range pos-x -1 1 pos-y -0.11 -9E-2
block edge apply stress 0.0 0.0 -10.0 range pos-x -1 1 pos-y 9E-2 0.11
block solve ratio 1.0E-5
model save 'direct10.sav'
```

```
;
; functions to calculate average joint stresses
; and average joint displacements
;
;Name:ini_jdisp
fish define ini_jdisp
   njdisp0 = 0.0
   sjdisp0 = 0.0
   ic = block.contact.head
   loop while ic # 0
      njdisp0 = njdisp0 + block.contact.disp.normal(ic)
      sjdisp0 = sjdisp0 + block.contact.disp.shear(ic)
      ic = block.contact.next(ic)
   endloop
end
@ini_jdisp
;
;Name:sstav
fish define sstav
   whilestepping
   sstav1 = 0.0
   nstav = 0.0
   njdisp = 0.0
   sjdisp = 0.0
   ncon = 0
   jl   = 0.2              ; joint length
   ic = block.contact.head
   loop while ic # 0
      ncon = ncon+1
      sstav1 = sstav1 + block.contact.force.shear(ic)
      nstav = nstav + block.contact.force.normal(ic)
      njdisp = njdisp + block.contact.disp.normal(ic)
      sjdisp = sjdisp + block.contact.disp.shear(ic)
      sh_disp = block.contact.disp.shear(ic)
      if sh_disp > 2.0e-4 then
         block.contact.prop(ic,'friction') = ...
           35.0 - 5.0*(sh_disp - 2.0e-4)/2.0e-4
         block.contact.prop(ic,'cohesion') = ...
           2.0 - 2.0*(sh_disp - 2.0e-4)/2.0e-4
      endif
      if sh_disp > 4.0e-4 then
         block.contact.prop(ic,'friction') = 30.0
         block.contact.prop(ic,'cohesion') = 0.0
      endif
      ic = block.contact.next(ic)
   endloop
```

```
    if ncon # 0
       sstav1 = sstav1 / jl
       nstav = nstav / jl
       njdisp = (njdisp-njdisp0) / ncon
       sjdisp = (sjdisp-sjdisp0) / ncon
    endif
    sstav = sstav1
end
@sstav
block contact reset displacement
hist reset
fish history @sstav
fish history @nstav
fish history @njdisp
fish history @sjdisp
history interval 1

;
; Apply shear load by imposing x-velocity on top block
block gridpoint apply velocity-x 1E-2 ...
  range pos-x -1E-2 0.21 pos-y -1E-2 0.1
model save 'direct11.sav'
;
block cycle 11500
model save 'direct12.sav'
```

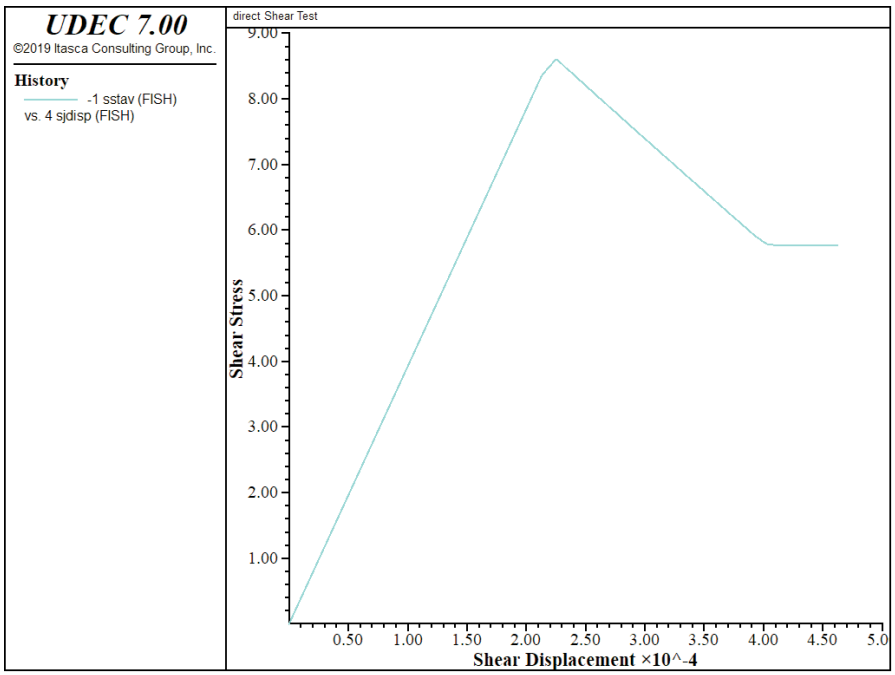

Figure 3.42 Average shear stress versus shear displacement – Coulomb joint strength controlled by FISH

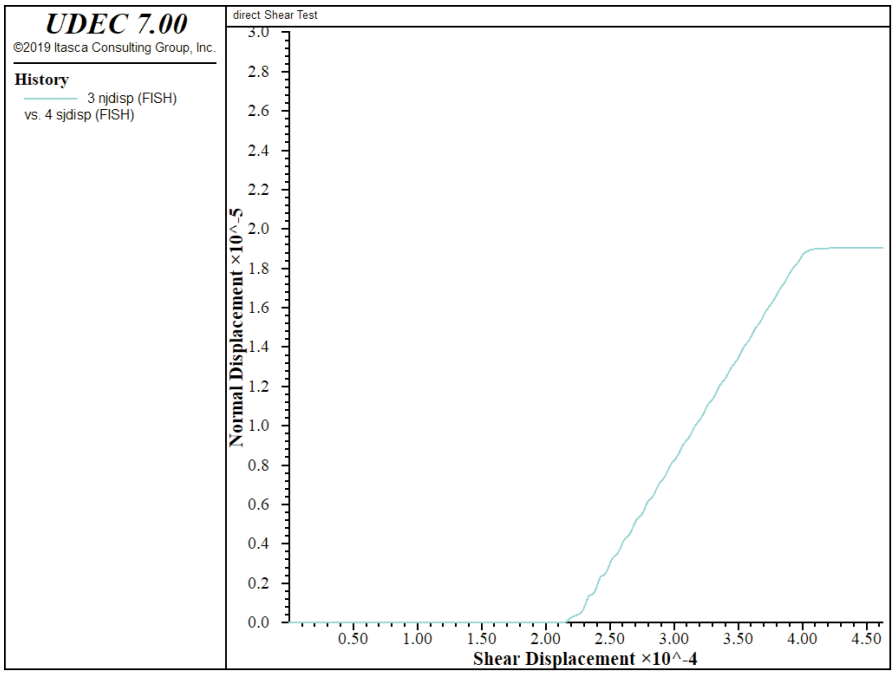

Figure 3.43 Average normal displacement versus shear displacement – Coulomb joint strength controlled by FISH

The displacement-weakening behavior is produced automatically with the continuously yielding joint model (**block contact cmodel assign cy**). This model simulates the progressive damage of the joint under shear. (For details, see Section 2 in **Constitutive Models**.) Example 3.25 contains the data file for the direct shear test using the continuously yielding model.

The test results using the continuously yielding model are shown in Figures 3.44 and 3.45. This model produces a smooth transition from peak to residual strength, and a gradual decrease in dilation as the joint is sheared.

Example 3.25 Direct shear test with continuously yielding joint model

```
model new
; create shear test model
block tolerance corner-round-length 0.001
block tolerance minimum-edge-length 0.002
block create polygon -5E-2,-0.1 -5E-2,0.1 0.25,0.1 0.25,-0.1
block cut crack (-1,0) (1,0)
```

```
block cut crack (0,0.1) (0,0)
block cut crack (0.2,0.1) (0.2,0)
block delete range pos-x -5E-2,0 pos-y 0,0.1
block delete range pos-x 0.2,0.25 pos-y 0,0.1
block zone gen quad 0.4,0.11 range pos-x 0,1 pos-y -1,0
block zone gen quad 0.07,0.11 range pos-x 0,1 pos-y 0,1
block zone group 'block'
block zone cmodel assign elastic density 2.6E-3 bulk 4.5E4 ...
  shear 3E4 range group 'block'
;
; C-Y block contact cmodel assign
block contact group 'cyjoint'
block contact cmodel assign cy friction 30 friction-initial 60 ...
  roughness 1E-4 stiffness-shear 4E4 stiffness-normal 4E4 ...
  range group 'cyjoint'
; new contact default
block contact cmodel default cy friction=30 friction-initial=60 ...
  roughness=1E-4 stiffness-shear=4E4 stiffness-normal=4E4
;
; Apply boundary conditions
bl grid app velocity-x 0 range pos-x -6E-2,-4E-2 pos-y -1,1
bl grid app velocity-x 0 range pos-x 0.24,0.26 pos-y -1,1
bl grid app velocity-y 0 range pos-x -1,1 pos-y -0.11,-9E-2
bl edg app stress 0.0,0.0,-10.0 range pos-x -1,1 pos-y 9E-2,0.11
block solve ratio 1.0E-5
model save 'direct13.sav'
;
; functions to calculate average joint stresses
; and average joint displacements
;
;Name:ini_jdisp
fish def ini_jdisp
   njdisp0 = 0.0
   sjdisp0 = 0.0
   ic = block.contact.head
   loop while ic # 0
      njdisp0 = njdisp0 + block.con.disp.normal(ic)
      sjdisp0 = sjdisp0 + block.con.disp.shear(ic)
      ic = block.con.next(ic)
   endloop
end
@ini_jdisp
;
;Name:av_str
fish def sstav
   whilestepping
```

```
    sstav1 = 0.0
    nstav = 0.0
    njdisp = 0.0
    sjdisp = 0.0
    ncon = 0
    jl   = 0.2              ; joint length
    ic = block.contact.head
    loop while ic # 0
       ncon = ncon+1
       sstav1 = sstav1 + block.con.force.shear(ic)
       nstav = nstav + block.con.force.normal(ic)
       njdisp = njdisp + block.con.disp.normal(ic)
       sjdisp = sjdisp + block.con.disp.shear(ic)
       ic = block.con.next(ic)
    endloop
    if ncon # 0
       sstav1 = sstav1 / jl
       nstav = nstav / jl
       njdisp = (njdisp-njdisp0) / ncon
       sjdisp = (sjdisp-sjdisp0) / ncon
    endif
    sstav = sstav1
end
@sstav
;
history reset
block contact reset disp
fish history @sstav
fish history @nstav
fish history @njdisp
fish history @sjdisp
history interval 1
;
; Apply shear load by imposing x-velocity on top block
bl grid app velocity-x 1E-2 range pos-x -1E-2,0.21 pos-y -1E-2,0.1
model save 'direct14.sav'
;
block cycle  18500
model save 'direct15.sav'
ret
```

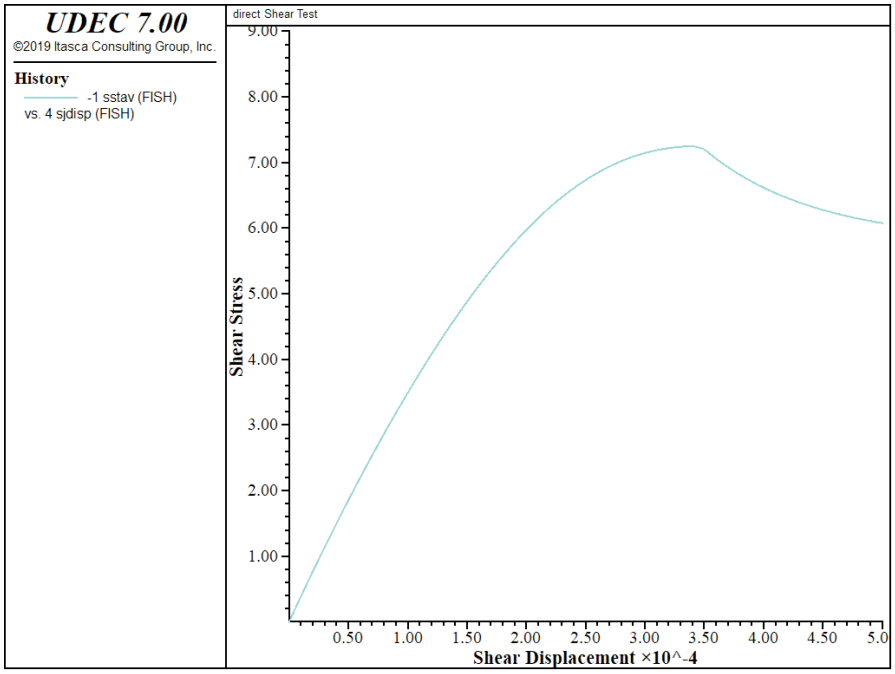

Figure 3.44 Average shear stress versus shear displacement – continuously yielding joint model

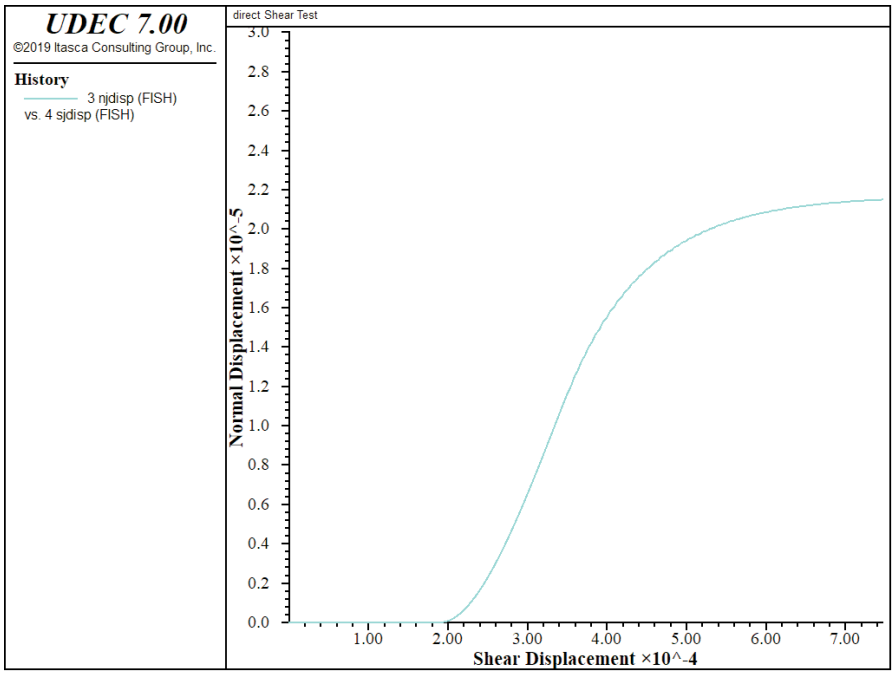

Figure 3.45 Average normal displacement versus shear displacement – continuously yielding joint model

The material properties for the Coulomb slip model and the continuously yielding joint model in these examples were selected to produce roughly the same response for joint shear strength and dilation for joints subjected to unidirectional shearing. For an actual application, properties should be selected (and adjusted as necessary) to simulate the response of the joints under the expected loading conditions. In most cases, the Coulomb model parameters are easier to estimate (see Section 3.8.2), and the simple modifications that are available with the Coulomb model may be sufficient to approximate the joint behavior.

For other joint models, such as the continuously yielding model and the Barton-Bandis joint model (see Section 3 in **Constitutive Models**), the determination of properties is more involved. In order to use the continuously yielding model, it is necessary to run a series of joint shear tests to best fit the model properties to physical test results. If Barton-Bandis index properties are available for the joints, then the Barton-Bandis joint model may be appropriate. It is recommended that simple shear tests *always* be performed, regardless of the joint model selected, to ensure that the joint behaves as expected under the anticipated problem conditions.

If it is necessary to simulate a complicated joint response, then a more complex joint model may be required. However, before going to a more complex model, it is usually helpful to apply a simple model first to establish a basis for evaluating the influence of the more complicated joint behavior. For example, several researchers describe a complex behavior for joints subjected to cyclic shearing in laboratory shear tests (e.g., see Jing et al. 1992, Huang et al. 1993, and Souley et al. 1995). These tests display two common features:

(1) change in shear strength upon load reversal; and

(2) recovery of joint dilation during reverse shearing.

A simple approximation of these effects can be obtained using the Coulomb model. The **block contact add-dil on** command (described earlier) adds the dilation angle to the input friction angle if the shear displacement increment is in the same direction as the total shear displacement, and subtracts the dilation angle if the shear displacement increment is in the opposite direction. We can run a cyclic shear test using the **block contact add-dil on** command with the data file in Example 3.26:

Example 3.26 Cyclic shear test with Coulomb slip model

```
model new
; create shear test model
block tolerance corner-round-length 0.001
block tolerance minimum-edge-length 0.002
block create polygon -5E-2,-0.1 -5E-2,0.1 0.25,0.1 0.25,-0.1
block cut crack (-1,0) (1,0)
block cut crack (0,0.1) (0,0)
block cut crack (0.2,0.1) (0.2,0)
block delete range pos-x -5E-2,0   pos-y 0,0.1
block delete range pos-x  0.2,0.25 pos-y 0,0.1
block zone gen quad 0.4,0.11 range pos-x 0,1  pos-y -1,0
block zone gen quad 0.07,0.11 range pos-x 0,1 pos-y 0,1
```

```
block zone group 'block'
block zone cmodel assign elastic density 2.6E-3 bulk 4.5E4 shear 3E4 ...
  range group 'block'
;Coulomb block contact cmodel assign with dilation added to friction
block contact group 'cycle joint'
block contact cmodel assign area stiffness-shear 1E6 ...
  stiffness-normal 1E6 friction 30 dilation 6 ...
  range group 'cycle joint'
; new contact default
block contact cmodel default area stiffness-shear=1E6 ...
  stiffness-normal=1E6 friction=30 dilation=6
block contact add-dilation on
;
; Apply boundary conditions
bl grid app velocity-x 0 range pos-x -6E-2,-4E-2 pos-y -1,1
bl grid app velocity-x 0 range pos-x 0.24,0.26 pos-y -1,1
bl grid app velocity-y 0 range pos-x -1,1 pos-y -0.11,-9E-2
bl edg app stress 0.0,0.0,-10.0 range pos-x -1,1 pos-y 9E-2,0.11
block solve ratio 1.0E-5
model save 'direct16.sav'
;
; functions to calculate average joint stresses
; and average joint displacements
;Name:ini_jdisp
fish def ini_jdisp
   njdisp0 = 0.0
   sjdisp0 = 0.0
   ic = block.contact.head
   loop while ic # 0
      njdisp0 = njdisp0 + block.con.disp.normal(ic)
      sjdisp0 = sjdisp0 + block.con.disp.shear(ic)
      ic = block.con.next(ic)
   endloop
end
@ini_jdisp
;Name:av_str
fish def sstav
   whilestepping
   sstav_ = 0.0
   nstav = 0.0
   njdisp = 0.0
   sjdisp = 0.0
   ncon = 0
   jl   = 0.2              ; joint length
   ic = block.contact.head
   loop while ic # 0
```

```
      ncon = ncon+1
      sstav_ = sstav_ + block.con.force.shear(ic)
      nstav = nstav + block.con.force.normal(ic)
      njdisp = njdisp + block.con.disp.normal(ic)
      sjdisp = sjdisp + block.con.disp.shear(ic)
      ic = block.con.next(ic)
   endloop
   if ncon # 0
      sstav_ = sstav_ / jl
      nstav = nstav / jl
      njdisp = (njdisp-njdisp0) / ncon
      sjdisp = (sjdisp-sjdisp0) / ncon
   endif
   sstav = sstav_
end
@sstav
;
block contact reset disp
history reset
fish history @sstav
fish history @nstav
fish history @njdisp
fish history @sjdisp
history interval 1
;
; Apply shear load by imposing x-velocity on top block
bl grid app velocity-x 1E-2 range pos-x -1E-2,0.21 pos-y -1E-2,0.11
model save 'direct17.sav'
block cycle  11500
model save 'direct18.sav'
;
; reverse shear load by imposing -xvelocity on top block
bl grid app velocity-x -1E-2 range pos-x -1E-2,0.21 pos-y -1E-2,0.11
block cycle 23000
model save 'direct19.sav'
;
;reverse shear load again
bl grid app velocity-x 1E-2 range pos-x -1E-2,0.21 pos-y -1E-2,0.11
block cycle  11500
model save 'direct20.sav'
```

The results of this test are shown in Figures 3.46 and 3.47. Both a change in peak shear strength upon load reversal and a recovery of dilation during reverse shearing are calculated with this model.

The joint behavior simulated by this model may be sufficient to evaluate the effect of cyclic loading. However, if the loading also produces a progressive damage of the joint, the Coulomb model may

not be appropriate because the model does not include this behavior. The continuously yielding model does simulate progressive damage, but does not model recovery of joint dilation during reverse shearing. The Barton-Bandis model may also be considered (see Section 3 in **Constitutive Models**). The modeler must then decide which model is most appropriate, or whether a new formulation must be developed.

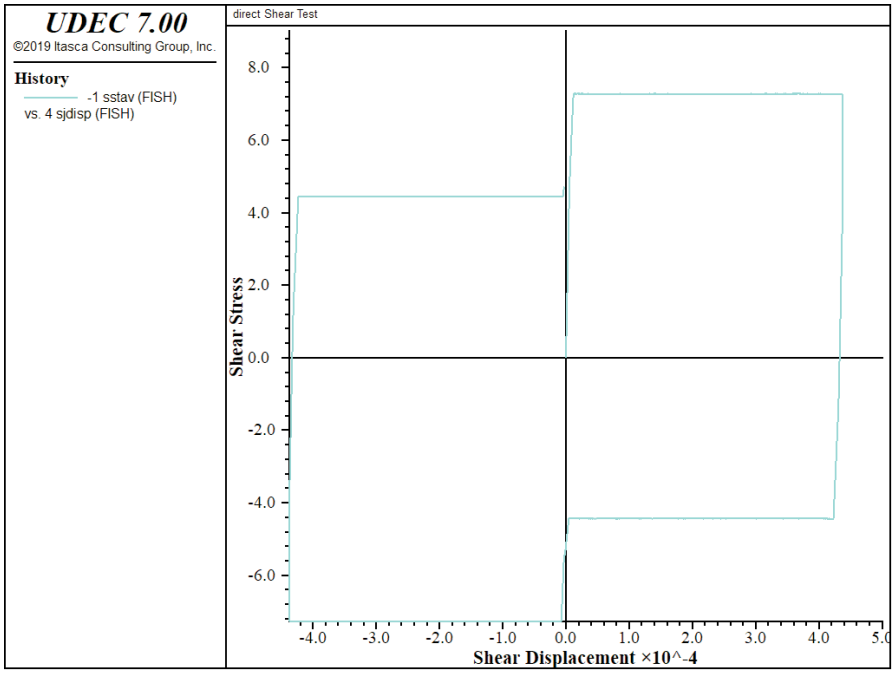

Figure 3.46 Average shear stress versus shear displacement for cyclic shear loading using the Coulomb slip model

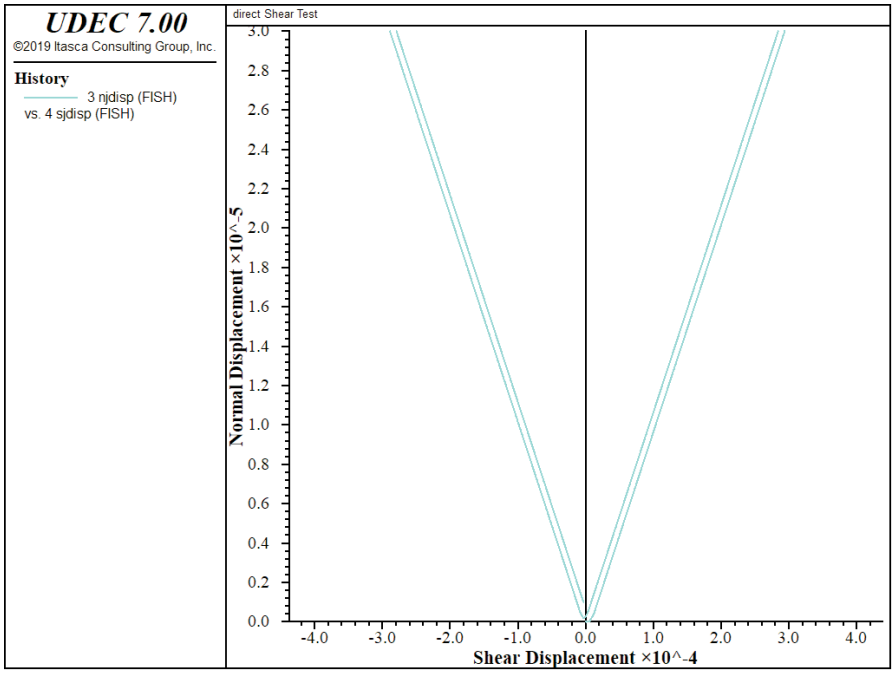

Figure 3.47 Average normal displacement versus shear displacement for cyclic shear loading using the Coulomb slip model

3.8 Material Properties

UDEC requires material properties both for the intact blocks and the discontinuities. This section provides an overview of typical properties used to represent the behavior of jointed rock, and presents guidelines for selecting the appropriate properties for a given model. There are also special considerations, such as the definition of post-failure properties and the extrapolation of laboratory-measured properties to the field scale. These topics are also discussed.

The selection of properties is often the most difficult element in the generation of a model because of the high uncertainty in the property database. When performing an analysis, especially in geomechanics, remember that the problem will always involve a data-limited system; the field data will never be known completely. However, with the appropriate selection of properties based upon the available database, important insight to the physical problem can still be achieved. This approach to modeling is discussed in more detail in Section 3.11.

3.8.1 Block Properties

Properties assigned to blocks are generally derived from laboratory testing programs. The following four sections describe intrinsic (laboratory-scale) properties and list common values for various rocks.

3.8.1.1 Mass Density

The mass density is required for every non-void material in a *UDEC* model. This property has units of mass divided by volume, and does *not* include the gravitational acceleration. In many cases, the *unit weight* of a material is prescribed. If the unit weight is given with units of force divided by volume, then this value must be divided by the gravitational acceleration before being entered as *UDEC* input for density.

3.8.1.2 Intrinsic Deformability Properties

All material models for deformable blocks in *UDEC* assume an isotropic material behavior in the elastic range described by two elastic constants: bulk modulus, K, and shear modulus, G. The elastic constants K and G (rather than Young's modulus, E, and Poisson's ratio, ν) are used in *UDEC* because it is believed that bulk and shear moduli correspond to more fundamental aspects of material behavior than do Young's modulus and Poisson's ratio. (See note 8 in Section 3.9 for justification of using (K,G) rather than (E,ν).)

The equations to convert from (E,ν) to (K,G) are

$$K = \frac{E}{3(1-2\nu)}$$
$$G = \frac{E}{2(1+\nu)} \tag{3.15}$$

Eq. (3.15) should not be used blindly when ν is near 0.5, since the computed value of K will be unrealistically high and convergence to the solution will be very slow. It is better to fix the value of K at its known physical value (estimated from an isotropic compaction test or from the p-wave speed), and then compute G from K and ν.

Some typical values for elastic constants are summarized in Table 3.4 for selected rocks.

Table 3.4 Selected elastic constants (laboratory-scale) for rocks (adapted from Goodman 1980)

	E (GPa)	ν	K (GPa)	G (GPa)
Berea sandstone	19.3	0.38	26.8	7.0
Hackensack siltstone	26.3	0.22	15.6	10.8
Bedford limestone	28.5	0.29	22.6	11.1
Micaceous shale	11.1	0.29	8.8	4.3
Cherokee marble	55.8	0.25	37.2	22.3
Nevada Test Site granite	73.8	0.22	43.9	30.2

3.8.1.3 Intrinsic Strength Properties

The basic criterion for block material failure in *UDEC* is the Mohr-Coulomb relation, which is a linear failure surface corresponding to shear failure:

$$f_s = \sigma_1 - \sigma_3 N_\phi + 2c\sqrt{N_\phi} \tag{3.16}$$

where $N_\phi = (1 + \sin\phi)/(1 - \sin\phi)$;
σ_1 = major principal stress (compressive stress is negative);
σ_3 = minor principal stress;
ϕ = friction angle; and
c = cohesion.

Shear yield is detected if $f_s < 0$. The two strength constants, ϕ and c, are conventionally derived from laboratory triaxial tests.

The Mohr-Coulomb criterion loses its physical validity when the normal stress becomes tensile but, for simplicity, the surface is extended into the tensile region to the point at which σ_3 equals the uniaxial tensile strength, σ^t. The minor principal stress can never exceed the tensile strength – i.e.,

$$f_t = \sigma_3 - \sigma^t \tag{3.17}$$

Tensile yield is detected if $f_t > 0$. Tensile strength for rock and concrete is usually derived from a Brazilian (or indirect tensile) test. Note that the tensile strength cannot exceed the value of σ_3, corresponding to the apex limit for the Mohr-Coulomb relation. This maximum value is given by

$$\sigma_{max}^{t} = \frac{c}{\tan \phi} \tag{3.18}$$

Typical values of cohesion, friction angle and tensile strength for a representative set of rock specimens are listed in Table 3.5:

Table 3.5 Selected strength properties (laboratory-scale) for rocks (adapted from Goodman 1980)

	friction angle (degrees)	cohesion (MPa)	tensile strength (MPa)
Berea sandstone	27.8	27.2	1.17
Repetto siltstone	32.1	34.7	—
Muddy shale	14.4	38.4	—
Sioux quartzite	48.0	70.6	—
Indiana limestone	42.0	6.72	1.58
Stone Mountain granite	51.0	55.1	—
Nevada Test Site basalt	31.0	66.2	13.1

Drucker-Prager strength parameters can be estimated from cohesion and friction angle properties. For example, assuming that the Drucker-Prager failure envelope circumscribes the Mohr-Coulomb envelope, the Drucker-Prager parameters q_ϕ and k_ϕ are related to ϕ and c by

$$q_\phi = \frac{6}{\sqrt{3}(3 - \sin \phi)} \sin \phi \tag{3.19}$$

$$k_\phi = \frac{6}{\sqrt{3}(3 - \sin \phi)} c \cos \phi \tag{3.20}$$

For further explanation on the relations between parameters, see Section 1 in **Constitutive Models**.

The ubiquitous-joint model also requires strength properties for the planes of weakness. Joint properties are discussed in Section 3.8.2. The properties for joint cohesion and friction angle also apply for the ubiquitous-joint model.

3.8.1.4 Post-Failure Properties

In many instances, particularly in mining engineering, the response of a material after the onset of failure is an important factor in the engineering design. Consequently, the post-failure behavior must be simulated in the material model. In *UDEC*, this is accomplished with properties that define four types of post-failure response:

(1) shear dilatancy;

(2) shear hardening/softening;

(3) volumetric hardening/softening; and

(4) tensile softening.

These properties are only activated after the onset of failure, as defined by the Mohr-Coulomb relation. Shear dilatancy is assigned for the Mohr-Coulomb, ubiquitous joint, strain-hardening / softening and double-yield models. Shear hardening/softening parameters are assigned for the strain-hardening/softening and double-yield models. Volumetric hardening/softening parameters are assigned for the double-yield models. Instantaneous tensile softening (the tensile strength is set to zero when tensile failure occurs) is prescribed in the Mohr-Coulomb and ubiquitous joint models.

Shear Dilatancy – Shear dilatancy, or dilatancy, is the change in volume that occurs with shear distortion of a material. Dilatancy is characterized by a dilation angle, ψ, which is related to the ratio of plastic volume change to plastic shear strain. This angle can be specified in the block plasticity models in *UDEC*. The dilation angle is typically determined from triaxial tests or shear box tests. For example, the idealized relation for dilatancy, based on the Mohr-Coulomb failure surface, is depicted for a triaxial test in Figure 3.48. The dilation angle is found from the plot of volumetric strain versus axial strain. Note that the initial slope for this plot corresponds to the elastic regime, while the slope used to measure the dilation angle corresponds to the plastic regime.

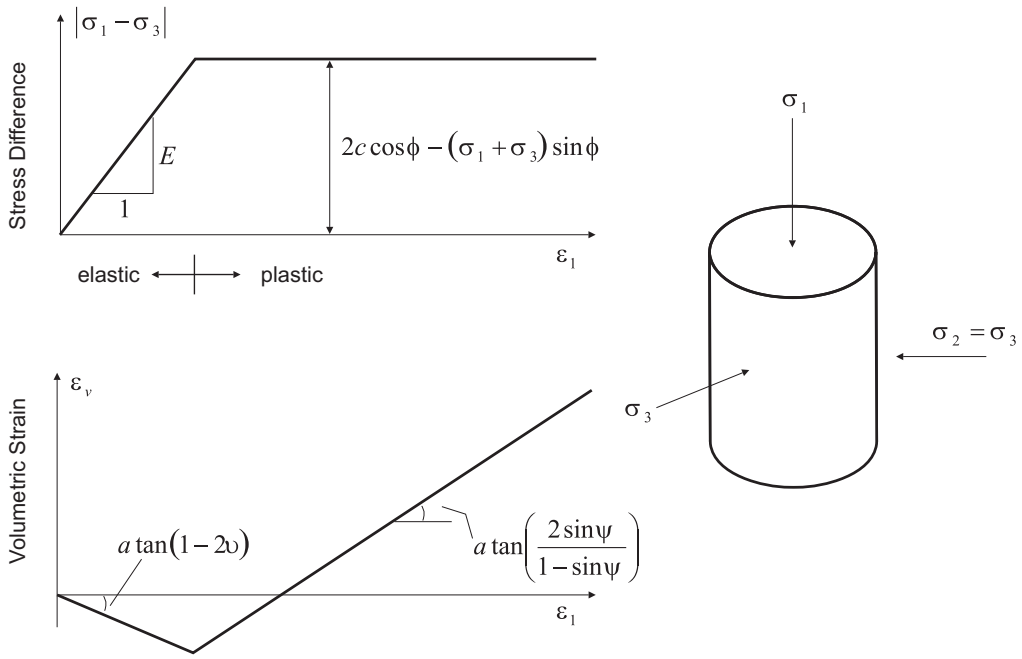

Figure 3.48 Idealized relation for dilation angle, ψ, from triaxial test results (Vermeer and de Borst 1984)

For soils, rocks and concrete, the dilation angle is generally significantly smaller than the friction angle of the material. Vermeer and de Borst (1984) report typical values for ψ:

dense sand	15°
loose sand	$< 10°$
normally consolidated clay	0°
granulated and intact marble	$12° - 20°$
concrete	12°

Vermeer and de Borst observe that values for the dilation angle are approximately between 0° and 20°, whether the material is soil, rock or concrete. The default value for dilation angle is zero for all the constitutive models in *UDEC*.

Dilation angle can also be prescribed for the joints in the ubiquitous-joint model. This property is typically determined from direct shear tests, and common values can be found in the references discussed in Section 3.8.2.

Shear Hardening/Softening – The initiation of material hardening or softening is a gradual process once plastic yield begins. At failure, deformation becomes more and more inelastic as a result of microcracking in concrete and rock, and particle sliding in soil. This also leads to degradation of strength in these materials and the initiation of shear bands. These phenomena, related to localization, are discussed in more detail in Section 3.11.

In *UDEC*, shear hardening and softening are simulated by making Mohr-Coulomb properties (cohesion and friction, along with dilation) functions of plastic strain (see Section 1 in **Constitutive Models**). These functions are accessed from the strain-softening and double-yield models, and can be specified by using the **TABLE** command.

Hardening and softening parameters must be calibrated for each specific analysis with values that are generally back-calculated from results of laboratory triaxial tests. This is usually an iterative process. Investigators have developed expressions for hardening and softening. For example, Vermeer and de Borst (1984) propose a frictional hardening relation:

$$\begin{aligned} \sin\phi_m &= 2\frac{\sqrt{e_p\, e_f}}{e_p + e_f}\ \sin\phi \quad \text{for } e_p \le e_f \\ \sin\phi_m &= \sin\phi \qquad\qquad\quad \text{for } e_p > e_f \end{aligned} \tag{3.21}$$

where ϕ = ultimate friction angle;
ϕ_m = mobilized friction angle;
e_p = plastic strain; and
e_f = hardening constant.

Numerical testing conditions can influence the model response for shear-hardening/softening behavior. The rate of loading can introduce inertial effects; this can be controlled by monitoring the unbalanced force and reducing the loading rate accordingly. A *FISH* function can be used to control the loading rate automatically. The results are also mesh-dependent; thus, it is important to evaluate the model behavior for differing zone sizes and mesh orientations whenever performing an analysis involving shear hardening or softening.

Volumetric Hardening/Softening – Volumetric hardening corresponds to irreversible compaction; increasing the isotropic pressure can cause permanent volume decrease. This behavior is common in materials such as lightly cemented sands and gravels, and hydraulically placed backfill.

Volumetric-hardening/softening may occur in the double-yield model. This model assumes that the hardening depends only on plastic volumetric strain.

The double-yield model takes its hardening rule from a table. Section 1.6.11.3 in **Constitutive Models** describes a recommended test procedure to develop these parameters from triaxial tests performed at constant mean stress, and for loading in which axial stress and confining pressure are kept equal. Typically, though, these data are not available. An alternative is to back-calculate the parameters from a uniaxial strain test.

Clark (1991) performed a series of uniaxial tests on mine backfill, and proposed a hardening rule for the volumetric yield cap. This can be written in a general form:

$$c_p = W\left[\frac{e_p^v}{H - e_p^v}\right]^\alpha + c_p^o \tag{3.22}$$

where c_p = the current cap pressure;
c_p^o = the initial cap pressure;
e_p^v = the plastic volumetric strain; and
H, W, α = parameters.

The experimental results of the uniaxial strain test by Clark (1991) are best fit for $H = 0.28$, $W = 1.15 \times 10^7$ and $\alpha = 1.5$. The relation in this case is

$$c_p = 1.15 \times 10^7 \left(\frac{e_p^v}{0.28 - e_p^v}\right)^{1.5} + 10^4 Pa \tag{3.23}$$

Tensile Softening – At the initiation of tensile failure, the tensile strength of a material will generally drop to zero. In the Zone Mohr-Coulomb model, the tensile strength is set to zero when tensile failure occurs in a zone (instantaneous softening) if flag-brittle is set true (default is false). The rate at which the tensile strength drops, or tensile softening occurs, can also be controlled by the plastic tensile strain in *UDEC*. This function is accessed from the strain-softening model and can be specified by using the **TABLE** command.

A simple tension test illustrates brittle tensile failure, as built into the Mohr-Coulomb model. The model is a tension test on a rectangular block composed of Mohr-Coulomb material. The ends of the sample are pulled apart at a constant velocity.

Example 3.27 Tension test on tensile-softening material

```
model new
; tension test model
model title 'tension test'
block tolerance corner-round-length 1E-2
block tolerance minimum-edge-length 2E-2
block create polygon 0 0 0 12 10 12 10 0
block zone gen edge 1.1
block zone group 'block'
block zone cmodel assign mohr-c density 2.5E3 bulk 1.19E10 ...
  shear 1.1E10 friction 44 cohesion 2.72E6 tension 2E6 ...
  flag-brittle on range group 'block'
block grid apply vel-y -1e-3 range pos-x 0 10 pos-y -0.1 0.1
block grid apply vel-y 1E-3 range pos-x 0 10 pos-y 11.9 12.1
block grid apply vel-x 0.0 range pos-x 0 10 pos-y -0.1 0.1
block grid apply vel-x 0.0 range pos-x 0 10 pos-y 11.9 12.1
```

```
; calculate average vertical stress and
; axial and orthogonal strain at center of model
;
fish define ax_str
  str = 0.0
  ib = block.boundary.head
  done = 0
  loop while done # 1
    ig = block.boundary.gp(ib)
    if ig >0 then
      if block.gp.pos.y(ig) > ytop then
        str = str - block.bou.force.y(ib) ;  axial force at top
      endif
    endif
    ib = block.boundary.next(ib)
    if ib = block.boundary.head then
      done = 1
    endif
  end_loop
  ib = block.head
  loop while ib # 0
    ig = block.gp(ib)
    loop while ig # 0
    count = count+1
      if block.gp.pos.y(ig) > ym_tol then
        if block.gp.pos.y(ig) < yp_tol then
          if block.gp.pos.x(ig) < xbot then
            xbpos = block.gp.pos.x(ig)
            xbdis = block.gp.disp.x(ig)   ;   x-disp. at left
          endif
          if block.gp.pos.x(ig) > xtop then
            xtpos = block.gp.pos.x(ig)
            xtdis = block.gp.disp.x(ig)   ;   x-disp. at right
          endif
        endif
      endif
      if block.gp.pos.x(ig) > xm_tol then
        if block.gp.pos.x(ig) < xp_tol then
          if block.gp.pos.y(ig) < ybot then
            ybpos = block.gp.pos.y(ig)
            ybdis = block.gp.disp.y(ig)   ;   y-disp. at bottom
          endif
          if block.gp.pos.y(ig) > ytop then
            ytpos = block.gp.pos.y(ig)
            ytdis = block.gp.disp.y(ig)   ;   y-disp. at top
          endif
```

```
        endif
      endif
      ig = block.gp.next(ig)
    end_loop
    ib = block.next(ib)
  end_loop
  ax_str = str / width
  ex_str = (xtdis - xbdis) / (xtpos - xbpos)
  ey_str = (ytdis - ybdis) / (ytpos - ybpos)
end
fish set @width=10 @ym_tol=5.9 @yp_tol=6.1
fish set @xbot=0.1 @xtop =9.9 @xm_tol=4.9
fish set @xp_tol=5.1 @ybot=0.1 @ytop=11.9
@ax_str
fish history @ax_str
fish history @ex_str
fish history @ey_str
block zone history stress-yy 5.0,6.0
block grid history displace-x 0.0,6.0
model save 'ten1.sav'
;
; damp auto
block cycle 10000
model save 'ten2.sav'
ret
```

The plot of axial stress versus axial strain (Figure 3.49) shows that the average axial stress across the top of the model drops to zero. The model decreases in width until the onset of tensile failure, then expands as tensile softening occurs (see Figure 3.50).

Note that the default damping (local damping) is suitable to minimize oscillations that can arise when the abrupt tensile failure occurs. Alternatively, if adaptive global damping is used by giving the **block mech damping global** command, oscillations are observed in the stress/strain plots. With adaptive global damping, the damping parameter is continually decreased as the model is stretched. When tensile failure occurs, the global damping parameter is low, and oscillations that may affect the final solution state are produced. With local damping, the amount of damping varies from gridpoint to gridpoint, and is proportional to the unbalanced force. This damping minimizes the oscillations that are produced when the abrupt tensile failure occurs. (See Section 1.2.7 in **Theory and Background** for further discussion on damping.)

The brittleness of the tensile softening can be controlled by the plastic tensile strain function, by using the strain-softening model instead of the Mohr-Coulomb model. As with the shear-softening model, the tensile-softening model must be calibrated for each specific problem and mesh size, since the results will be mesh-dependent.

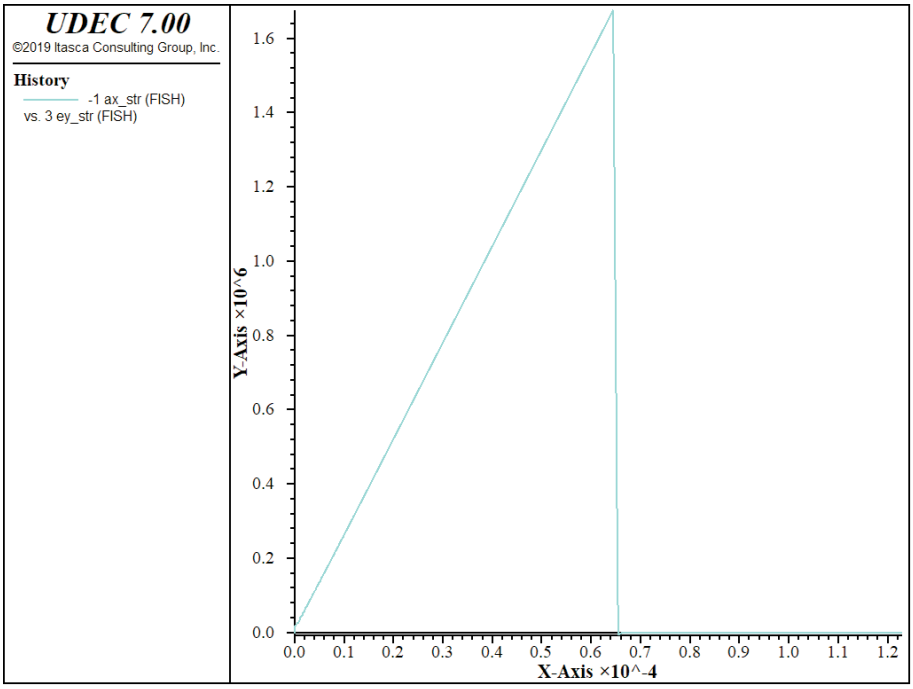

Figure 3.49 Axial stress versus axial strain for tensile test of tension softening material

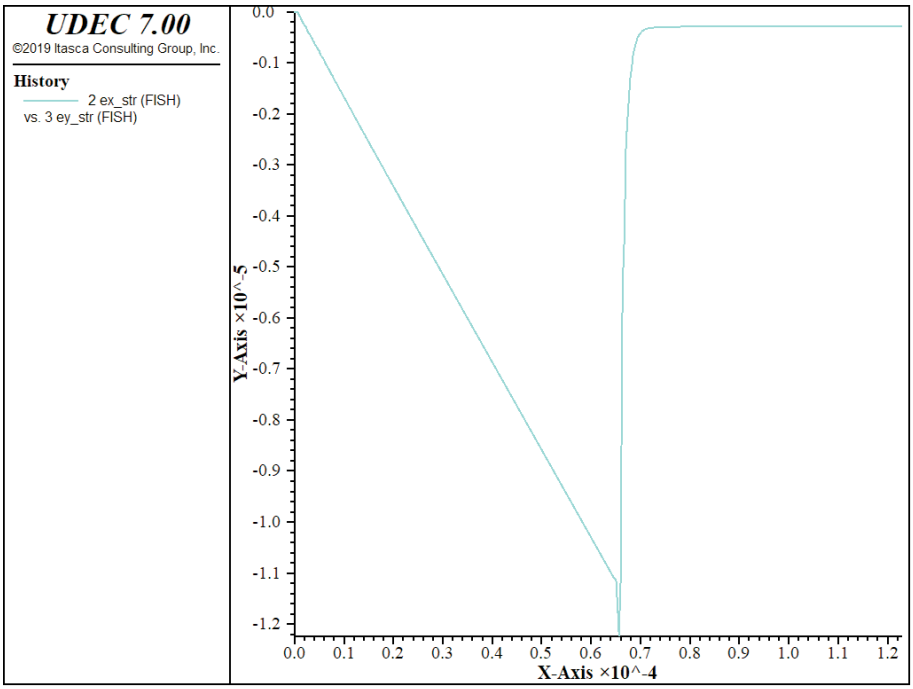

Figure 3.50 Orthogonal strain versus axial strain for tensile test of tension softening material

3.8.1.5 Extrapolation to Field-Scale Properties

The material properties used in the *UDEC* model should correspond as closely as possible to the actual values of the physical problem. Laboratory-measured properties generally should not be used directly in a *UDEC* model for a full-scale problem. The presence of discontinuities in the model will account for a good portion of the scaling effect on properties. However, some adjustment of block properties will probably still be required to represent the influence of heterogeneities and micro-fractures, fissures and other small discontinuities on the rock mass response.

Several empirical approaches have been proposed to derive field-scale properties. Some of the more commonly accepted methods are discussed.

Deformability of a rock mass is generally defined by a modulus of deformation, E_m. If the rock mass contains a set of relatively parallel, continuous joints with uniform spacing, the value for E_m can be estimated by treating the rock mass as an equivalent transversely isotropic continuum. The relations in Section 3.8.2 can then be used to estimate E_m in the direction normal to the joint set. Deformation moduli can also be estimated for cases involving more than one set of discontinuities. The references listed in Section 3.8.2 provide solutions for multiple joint sets.

In practice, the rock mass structure is often much too irregular, or sufficient data are not available, to use the above approach. It is common to determine E_m from a force-displacement curve obtained from an in-situ compression test. Such tests include plate bearing tests, flatjack tests and dilatometer tests.

Bieniawski (1978) developed an empirical relation for E_m, based on field test results at sites throughout the world. The relation is based on rock mass rating (RMR). For rocks with a rating higher than 55, the test data can be approximately fit to

$$E_m = 2(\text{RMR}) - 100 \tag{3.24}$$

The units of E_m are GPa.

For values of E_m between 1 and 10 GPa, Serafim and Pereira (1983) found a better fit, given by

$$E_m = 10^{\frac{\text{RMR}-10}{40}} \tag{3.25}$$

References by Goodman (1980) and Brady and Brown (1985) provide additional discussion on these methods.

The most commonly accepted approach to estimate rock mass strength is that proposed by Hoek and Brown (1980). They developed the empirical rock mass strength criterion,

$$\sigma_{1s} = \sigma_3 + (m\sigma_c\sigma_3 + s\sigma_c^2)^{1/2} \tag{3.26}$$

where σ_{1s} = major principal stress at peak strength;
σ_3 = minor principal stress;
m and s = constants that depend on the properties of the rock and the extent to which it has been broken before being subjected to failure stresses; and
σ_c = uniaxial compressive strength of intact rock material.

The unconfined compressive strength for a rock mass is given by

$$q_m = \sigma_c \, s^{1/2} \tag{3.27}$$

and the uniaxial tensile strength of a rock mass is

$$\sigma_t = \frac{1}{2} \sigma_c \, [m - (m^2 + 4s)^{1/2}] \tag{3.28}$$

Table 3.6, from Hoek and Brown (1988), presents typical values for m and s for undisturbed and disturbed rock masses.

It is possible to estimate Mohr-Coulomb friction angle and cohesion from the Hoek-Brown criterion (see, for example, Hoek 1990).

For a given value of σ_3, a tangent to the function (Eq. (3.26)) will represent an equivalent Mohr-Coulomb yield criterion in the form

$$\sigma_1 = N_\phi \sigma_3 + \sigma_c^M \tag{3.29}$$

where $N_\phi = \frac{1+\sin\phi}{1-\sin\phi} = \tan^2 (\frac{\phi}{2} + 45°)$

By substitution, σ_c^M is

$$\sigma_c^M = \sigma_1 - \sigma_3 N_\phi = \sigma_3 + \sqrt{\sigma_3 \sigma_c m + \sigma_c^2 s} - \sigma_3 N_\phi = \sigma_3 (1 - N_\phi) + \sqrt{\sigma_3 \sigma_c m + \sigma_c^2 s}$$

σ_c^M is the apparent uniaxial compressive strength of the rock mass for that value of σ_3.

The tangent to the Eq. (3.26) is defined by

$$N_\phi(\sigma_3) = \frac{\partial \sigma_1}{\partial \sigma_3} = 1 + \frac{\sigma_c m}{2\sqrt{\sigma_3 \cdot \sigma_c m + s\sigma_c^2}} \tag{3.30}$$

The cohesion (c) and friction angle (ϕ) can then be obtained from N_ϕ and σ_c^M:

$$\phi = 2\tan^{-1}\sqrt{N_\phi} - 90^\circ \tag{3.31}$$

$$c = \frac{\sigma_c^M}{2\sqrt{N_\phi}} \tag{3.32}$$

Table 3.6 Typical values for Hoek-Brown rock-mass strength parameters (adapted from Hoek and Brown 1988)

Disturbed rock mass m and s values		Undisturbed rock mass m and s values				
EMPIRICAL FAILURE CRITERION $\sigma'_1 = \sigma'_3 + \sqrt{(m\sigma_c\sigma'_3 + s\sigma^2_c)}$ σ'_1 = major principal effective stress σ'_3 = minor principal effective stress σ_c = uniaxial compressive strength of intact rock, and m and s are empirical constants.		CARBONATE ROCKS WITH WELL-DEVELOPED CRYSTAL CLEAVAGE — dolomite, limestone and marble	LITHIFIED ARGILLACEOUS ROCKS — mudstone, siltstone, shale and slate (normal to cleavage)	ARENACEOUS ROCKS WITH STRONG CRYSTALS AND POORLY DEVELOPED CRYSTAL CLEAVAGE — sandstone and quartzite	FINE-GRAINED POLYMINERALLIC IGNEOUS CRYSTALLINE ROCKS — andesite, dolerite, diabase and rhyolite	COARSE-GRAINED POLYMINERALLIC IGNEOUS & METAMORPHIC CRYSTALLINE ROCKS — amphibolite, gabbro gneiss, granite, norite, quartz-diorite
INTACT ROCK SAMPLES						
Laboratory specimens free	m	7.00	10.00	15.00	17.00	25.00
from discontinuities	s	1.00	1.00	1.00	1.00	1.00
CSIR rating: RMR = 100	m	7.00	10.00	15.00	17.00	25.00
NGI rating: Q = 500	s	1.00	1.00	1.00	1.00	1.00
VERY GOOD QUALITY ROCK MASS						
Tightly interlocking undisturbed rock	m	2.40	3.43	5.14	5.82	8.56
with unweathered joints at 1 to 3 m	s	0.082	0.082	0.082	0.082	0.082
CSIR rating RMR = 85	m	4.10	5.85	8.78	9.95	14.63
NGI rating: Q = 100	s	0.189	0.189	0.189	0.189	0.189
GOOD QUALITY ROCK MASS						
Fresh to slightly weathered rock, slightly	m	0.575	0.821	1.231	1.395	2.052
disturbed with joints at 1 to 3 m	s	0.00293	0.00293	0.00293	0.00293	0.00293
CSIR rating: RMR = 65	m	2.006	2.865	4.298	4.871	7.163
NGI rating: Q = 10	s	0.0205	0.0205	0.0205	0.0205	0.0205
FAIR QUALITY ROCK MASS						
Several sets of moderately weathered	m	0.128	0.183	0.275	0.311	0.458
joints spaced at 0.3 to 1 m	s	0.00009	0.00009	0.00009	0.00009	0.00009
CSIR rating: RMR = 44	m	0.947	1.353	2.03	2.301	3.383
NGI rating: Q = 1	s	0.00198	0.00198	0.00198	0.00198	0.00198
POOR QUALITY ROCK MASS						
Numerous weathered joints at 30-500 mm,	m	0.029	0.041	0.061	0.069	0.102
some gouge; clean compacted waste rock	s	0.000003	0.000003	0.000003	0.000003	0.000003
CSIR rating: RMR = 23	m	0.447	0.639	0.959	1.087	1.598
NGI rating: Q = 0.1	s	0.00019	0.00019	0.00019	0.00019	0.00019
VERY POOR QUALITY ROCK MASS						
Numerous heavily weathered joints spaced	m	0.007	0.01	0.015	0.017	0.025
<50 mm with gouge; waste rock with fines	s	0.0000001	0.0000001	0.0000001	0.0000001	0.0000001
CSIR rating: RMR = 3	m	0.219	0.313	0.469	0.532	0.782
NGI rating: Q = 0.01	s	0.00002	0.00002	0.00002	0.00002	0.00002

3.8.2 Joint Properties

Joint properties are conventionally derived from laboratory testing (e.g., triaxial and direct shear tests). These tests can produce physical properties for joint friction angle, cohesion, dilation angle and tensile strength, as well as joint normal and shear stiffnesses. The joint cohesion and friction angle correspond to the parameters in the Coulomb strength criterion.

Values for normal and shear stiffnesses for rock joints typically can range from roughly 10 to 100 MPa/m (for joints with soft clay in-filling), to over 100 GPa/m (for tight joints in granite and basalt). Published data on stiffness properties for rock joints are limited; summaries of data can be found in Kulhawy (1975), Rosso (1976) and Bandis et al. (1983).

Approximate stiffness values can be back-calculated from information on the deformability and joint structure in the jointed rock mass and the deformability of the intact rock. If the jointed rock mass is assumed to have the same deformational response as an equivalent elastic continuum, then relations can be derived between jointed rock properties and equivalent continuum properties.

For uniaxial loading of rock containing a single set of uniformly spaced joints oriented normal to the direction of loading, the following relation applies.

$$\frac{1}{E_m} = \frac{1}{E_r} + \frac{1}{k_n s}$$

or

$$k_n = \frac{E_m \, E_r}{s \, (E_r - E_m)} \tag{3.33}$$

where E_m = rock mass Young's modulus;

E_r = intact rock Young's modulus;

k_n = joint normal stiffness; and

s = joint spacing.

A similar expression can be derived for joint shear stiffness:

$$k_s = \frac{G_m \, G_r}{s \, (G_r - G_m)} \tag{3.34}$$

where G_m = rock mass shear modulus;

G_r = intact rock shear modulus; and

k_s = joint shear stiffness.

The equivalent continuum assumption, when extended to three orthogonal joint sets, produces the following relations.

$$E_i = \left(\frac{1}{E_r} + \frac{1}{s_i \, k_{ni}} \right)^{-1} \qquad (i = 1,\ 2,\ 3) \tag{3.35}$$

$$G_{ij} = \left(\frac{1}{G_r} + \frac{1}{s_i \, k_{si}} + \frac{1}{s_j \, k_{sj}} \right)^{-1} \qquad (i\ ,\ j = 1,\ 2,\ 3) \tag{3.36}$$

Several expressions have been derived for two- and three-dimensional characterizations and multiple joint sets. References for these derivations can be found in Singh (1973), Gerrard (1982(a) and (b)) and Fossum (1985).

There is a limit to the maximum joint stiffnesses that are reasonable to use in a *UDEC* model. If the physical normal and shear stiffnesses are less than ten times the equivalent stiffness of adjacent zones (see Eq. (3.37) in Section 3.9), then there is no problem in using physical values. If the ratio is more than ten, the solution time will be significantly longer than for the case in which the ratio is limited to ten, without much change in the behavior of the system. Serious consideration should be given to reducing supplied values of normal and shear stiffnesses to improve solution efficiency. There may also be problems with block interpenetration if the normal stiffness, k_n, is very low. A rough estimate should be made of the joint normal displacement that would result from the application of typical stresses in the system ($u = \sigma / k_n$). This displacement should be small compared to a typical zone size. If it is greater than, say, 10% of an adjacent zone size, then either there is an error in one of the numbers, or the stiffness should be increased.

Published strength properties for joints are more readily available than stiffness properties. Summaries can be found, for example, in Jaeger and Cook (1969), Kulhawy (1975) and Barton (1976). Friction angles can vary from less than 10° for smooth joints in weak rock (such as tuff) to over 50° for rough joints in hard rock (such as granite). Joint cohesion can range from zero cohesion to values approaching the compressive strength of the surrounding rock.

It is important to recognize that joint properties measured in the laboratory typically are not representative of those for real joints in the field. Scale dependence of joint properties is a major question in rock mechanics. Often, the only way to guide the choice of appropriate parameters is by comparison to similar joint properties derived from field tests. However, field tests observations are extremely limited. Some results are reported by Kulhawy (1975).

3.9 Tips and Advice

When problem solving with *UDEC*, it is often important to try to optimize the model for the most efficient analysis. This section provides several suggestions on ways to improve a model run. Also, some common pitfalls that should be avoided when preparing a *UDEC* calculation are listed.

1. **Selecting Joint Geometry**

   The selection of the joint geometry for input to a *UDEC* model is a crucial step in the analysis. In a multiple-jointed rock mass with tens to hundreds of mapped joint traces, it is not feasible to incorporate all identified joint structures. Typically, only a very small percentage of joint locations can actually be input, in order to create models of reasonable size and execution speed for practical analysis. Thus, the modeler must filter joint geometry data and select only those joints that are critical to the mechanical response. This is usually an iterative process, often beginning with borehole and trace-map fracture data. The difficulty in selecting critical fractures centers around how statistical data are represented in the deterministic *UDEC* model. For multiple joint sets, it should be assumed that orientations of joints within a given set are the same, and that small variations (say, less than 10° to 15°) may be neglected. Often, the jointed rock region can be divided into subregions delimited by specific joints, and all joints within a subregion are considered continuous. This is an upper-bound estimate for effects of variation in trace length and persistence.

   The next step is to use a filtering criterion to identify those joints most susceptible to slip or separation for the prescribed loading conditions. This criterion may range from: (1) identifying whether sufficient degrees of freedom are provided to allow slip; to (2) utilizing principles of block theory to check the kinematics of joint displacement; to (3) comparison with in-situ observation and records (for example, from microseismic monitoring) to identify critical joints.

2. **Designing the Model**

   It is tempting to try to build as much detail of the geologic structure as possible into a *UDEC* model. There are three main arguments against this approach: (1) it is futile to ever expect to have sufficient data to model a jointed rock mass in every detail; (2) the computer hardware requirements for a detailed model quickly exceed that typically available for engineering projects; and (3) most importantly, a controlled engineering understanding of model results becomes less effective as more detail is added.

   Two considerations should be kept in mind when creating the *UDEC* model. The first is whether or not a discontinuum analysis is actually required. This depends in large part on the ratio of the scale of the physical system under investigation to the average spacing of the joint structure. For example, if an excavation is made in a rock mass containing a single joint set with an average spacing of 1 m or less, and the minimum dimension of the excavation

is 10 m, then a continuum analysis with a ubiquitous-joint material model may be a more reasonable approach. At this scale, the continuum analysis can produce a response that is broadly equivalent to that obtained when the joints are explicitly modeled. The analysis with *UDEC* provides a more detailed analysis of the failure mechanism, but may require more computation time than the continuum analysis. In instances where the ratio of the physical system scale to the joint spacing exceeds approximately 10:1, a continuum analysis may be preferred. Analyses should be conducted with both continuum and discontinuum analyses when there is a question as to whether the continuum analysis is sufficient to represent the discontinuum response. See Board et al. (1996) for an illustration of the application of discontinuum and continuum analyses (using *UDEC* and *FLAC*) in a comparative study of toppling behavior of a jointed rock slope.

The second consideration is the extent to which the detailed geologic structure should be included in the model. A detailed representation that includes the most critical joint structure (see note 1, above) is usually only required within a limited region surrounding the area of interest (e.g., within a few tunnel radii surrounding a tunnel excavation). Generally, the greater jointing detail only needs to extend from the area of interest to a distance sufficient to encompass the region in which failure is anticipated. That is, the detailed geologic structure should extend beyond the distance to which joint slip and separation are calculated.

3. Check Model Runtime

The solution time for a *UDEC* run is a function of both the number of rigid blocks or gridpoints in deformable blocks, and the number of contacts in a model. If there are very few contacts in the model, then the time is proportional to $N^{3/2}$, where N is the number of rigid blocks or gridpoints in deformable blocks. This formula holds for elastic problems. The runtime will vary somewhat, but not substantially, for plasticity problems.

The solution time will increase as more contacts are created in the model. It is important to check the speed of calculation on your computer for a specific model. An easy way to do this is to run the benchmark test described in Section 5.1. Then use this speed to estimate the speed of calculation for the specific model, based on interpolation from the number of gridpoints and contacts.

4. Effects on Runtime

UDEC will take a longer time to converge if:

(a) there are large contrasts in stiffness in block materials, or in joint materials, or in block versus joint materials; or

(b) there are large contrasts in block or zone sizes.

The code becomes less efficient as these contrasts become greater. The effect of a contrast in stiffness should be investigated before performing a detailed analysis. For example, for mechanical-only calculations, joint normal and shear stiffnesses should be kept smaller than ten times the equivalent stiffness of the stiffest neighboring zone in blocks adjoining the joint – i.e.,

$$k_n \text{ and } k_s \leq 10.0 \left[max \left[\frac{K + 4/3G}{\Delta z_{\text{min}}} \right] \right] \tag{3.37}$$

where K and G are bulk and shear moduli, respectively, of the block material, and Δz_{min} is the smallest width of the zone adjoining the joint in the normal direction. If the joint stiffnesses are greater than 10 times the equivalent stiffness, the solution time of the model will be significantly longer than for the case in which the ratio is limited to ten, without a significant change in the behavior of the system.

On the other hand, there may be problems if the normal stiffness, k_n, is very low. A rough estimate should be made of the joint normal displacement that would result from the application of typical stresses in the system ($u = \sigma / k_n$). This displacement should be small compared to a typical zone size. If it is greater than roughly 10% of an adjacent zone size, then either there is an error in one of the numbers, or the stiffness should be increased.

5. Considerations for Density of Zoning

UDEC uses constant-strain elements in deformable blocks. If the gradient of stress/strain is high, many zones will be needed to represent the varying distribution. Run the same problem with different zoning densities to check the effect. Constant-strain zones are used in *UDEC* because a better accuracy is achieved, when modeling plastic flow, with many low-order elements than with a few high-order elements (see Section 1.2.5 in **Theory and Background**).

Try to keep the zoning as uniform as possible, particularly in the region of interest. Avoid long, thin zones with aspect ratio greater than 5:1.

6. Check Model Response

UDEC shows how a system behaves. Make frequent simple tests to make sure that you are doing what you think you are doing. For example, if a loading condition and geometry are symmetrical, make sure that the response is symmetrical. After making a change in the model, execute a few calculation steps (say, 5 or 10) to verify that the initial response is of the correct sign and in the correct location. Do back-of-the-envelope estimates of the expected order of magnitude of stress or displacements, and compare to the *UDEC* output.

If you apply a violent shock to the model, you will get a violent response. If you do physically unreasonable things to the model, you must expect strange results. If you get unexpected results at a given stage of an analysis, review the steps you followed up to this stage.

Critically examine the output before proceeding with the model simulation. If, for example, everything appears reasonable except for large velocities in one corner block, do not go on until you understand the reason. In this case, you may have not fixed a boundary corner properly.

7. Check Block Contacts

When generating complex joint patterns, the **block cycle 0** command can be used, followed plotting contacts. Ensure that all contacts exist between blocks, and that a block does not contact itself. For example, the following data file connects a circular tunnel to the exterior of the model but, in doing so, creates a block in contact with itself:

```
block tol bl-ro-len 0.1
bl create poly 0,0  0,10  10,10  10,0
bl cut tunnel 5,5  2  16
bl cut crack 0,5  3,5
```

The block surrounding the tunnel is in contact with itself; consequently, all contacts are not created. The **bloc cut crack** command should extend completely across the model to prevent this.

8. Use Bulk and Shear Moduli

It is better to use bulk modulus, K, and shear modulus, G, than Young's modulus, E, and Poisson's ratio, ν, for elastic properties in *UDEC*.

The pair (K, G) makes sense for all elastic materials that do not violate thermodynamic principles. The pair (E, ν) does not make sense for certain admissible materials. At one extreme, we have materials that resist volumetric change but not shear; and at the other extreme, materials that resist shear but not volumetric change. The first type of material corresponds to finite K and zero G, and the second to zero K and finite G. However, the pair (E, ν) is not able to characterize either the first or the second type of material. If we exclude the two limiting cases (conventionally, $\nu = 0.5$ and $\nu = -1$), the equations

$$\begin{aligned} 3K(1-2\nu) &= E \\ 2G(1+\nu) &= E \end{aligned} \tag{3.38}$$

relate the two sets of constants. These equations hold, however close we approach (but not reach) the limiting cases. We do not need to relate them to physical tests that may or may not be feasible – the equations are simply the consequence of two possible ways of defining coefficients of proportionality. Suppose we have a material in which the resistance to distortion progressively reduces, but in which the resistance to volume change remains constant. ν approaches 0.5 in this case. The equation $3K(1-2\nu) = E$ must still be satisfied. There are two possibilities (argued on *algebraic*, not physical, grounds): either E remains finite (and nonzero) and K tends to an arbitrarily large value, or K remains finite and E tends to zero. We rule out the first possibility because there is a limiting compressibility to all known materials (e.g., 2 GPa for water, which has a Poisson's ratio of 0.5). This leaves the second, in which E is varying drastically, even though we supposed that the material's principal mode of elastic resistance was unchanging. We deduce that the parameters (E, ν) are inadequate to express the material behavior.

9. Choice of Damping

In most instances, it is recommended that **block mech damping local** be used for static analyses. This is the default damping mode in *UDEC*, and is generally appropriate for static analysis for the reasons given in Section 1.2.7 in **Theory and Background**. Also, as demonstrated in Example 3.27, local damping is more suitable to minimize oscillations that may arise when abrupt failure occurs in the model.

In some cases, particularly when calculating an initial equilibrium state, it may be more computationally efficient to use **block mech damping global**. As discussed in Section 1 in **Theory and Background**, local damping is most efficient when velocity components at gridpoints pass through zero periodically, because the mass-adjustment process depends on velocity sign changes. Adaptive global damping, on the other hand, applies a constant damping factor that is not affected by velocity sign changes. If velocities act predominantly in one direction (e.g., due to gravity loading), then a system with local damping may take longer to converge than one with adaptive global damping. When in doubt, it is usually best to run the model with both **block mech damping local** and **block mech damping global**, and compare the calculation steps required to reach convergence.

10. Assigning Block and Joint Models and Properties

The **block zone cmodel assign** and **block contact cmodel assign** commands are provided to allow users to assign model properties directly to zones and contacts. These commands should be used in general, and are necessary if there is a variation in properties in a model, or if the user wishes to control the property variation with *FISH*.

If an extremely large model that requires an extensive amount of memory is created, then the **block change** and **block property** commands can be used

to reduce the amount of memory needed, provided that there is little or no variation in properties in the model. Model properties are assigned to property numbers for these commands, and therefore less memory is required than for the **block zone cmodel assign** and **block contact cmodel assign** commands.

11. Contact Overlap

The error message "Contact Overlap Too Great" can occur during cycling if one block penetrates too far into another. The maximum amount of overlap allowed by the code is one-half the rounding length. If such an error occurs, it is usually necessary to restart the problem from an earlier state. However, before restarting, it is important to identify the cause of the error and correct it. Useful information concerning the contact location(s) where overlap occurred is given preceding the contact overlap message. Also, select the overlap attribute in a block plot to identify the blocks involved. The following possible causes for contact overlap can be identified.

Joint Normal Stiffness Too Low – If the joint normal stiffness is unrealistically low for the loads applied, the blocks will penetrate too far into each other. This cause can often be identified by (a) plotting a close-up of the affected area; or (b) printing contacts in the same area. See note 4, above, for a remedy to this problem.

Numerical Instability – Numerical instabilities, characterized by increasing amplitude of oscillations, result from timesteps that are too large. History plots that show wild fluctuations are indicative of a numerical instability. The *only* way to correct a numerical instability without changing other problem parameters is to reduce the timestep by using the **block mechanical timestep factor** command. Increasing mass damping parameters can often hide instability but will not likely eliminate it. *UDEC* automatically determines a timestep that is stable for most cases. However, situations may arise when this timestep is too large. Several situations have been identified as causing numerical instability:

(1) use of large values of stiffness proportional damping at high frequencies (see Section 4.2 in **Special Features**);

(2) use of high values of joint dilation;

(3) use of problem geometries in which one block contacts many (more than 3) other blocks on one side; and

(4) use of nonreflecting (i.e., viscous) boundaries in which the bounding material is significantly stiffer than the material in the problem domain.

If the cause of the contact overlap cannot be identified, it may be necessary to use the **SET cscan** command. This command causes the location of the center of rounded corners to be updated more frequently, resulting in more accurate calculation of contacts and overlap.

Another procedure for eliminating this error is to increase the overlap tolerance at the start of the problem. Use the command **block contact tolerance overlap**. This procedure is useful if the original rounding length was very small and/or the problem geometry involved blocks with very acute angles. However, this approach can also be very dangerous if not used thoughtfully, because erroneous results can occur if the overlap is too large.

12. Disconnected Blocks

The data structure of *UDEC* was designed specifically to model compact rock masses (i.e., tightly packed blocks). The code can handle isolated cases where blocks lose contact with their neighbors by using "virtual contacts," which are fictitious links between particles. However, the code is not designed for multiple virtual contacts; in this case, it becomes inefficient. It is suggested that these disconnected blocks be deleted if they are of no further interest. For analyses involving many disconnected blocks, use the **block config cell** option.

13. Initializing Variables

It is common practice to initialize the displacements of the gridpoints and joint contacts (**block mechanical reset disp joint-disp**) between runs to aid in the interpretation of a simulation in which many different excavation stages are performed. This can be done for gridpoints because the code does not require the displacements in the calculation sequence – they are determined from the velocities of the gridpoints as a convenience to the user. For contacts, resetting shear displacements may affect results if the joints dilate. Contact displacements are stored in duplicate so that values can be reset for printing and plotting, without affecting model results.

14. Determining Collapse Loads

In order to determine a collapse load, it often is better to use "strain-controlled" rather than "stress-controlled" boundary conditions (i.e., apply a constant velocity and measure the boundary reaction forces rather than apply forces and measure displacements). A system that collapses becomes difficult to control as the applied load approaches the collapse load. This is true of a real system as well as a model system. (See Section 8 in the **Verification Problems**.)

3.10 Interpretation

Because *UDEC* models a nonlinear system as it evolves in time, the interpretation of results may be more difficult than with a conventional finite element program that produces a "solution" at the end of its calculation phase. There are several indicators that can be used to assess the state of the numerical model for a static analysis (e.g., whether the system is stable, unstable or is in steady-state plastic flow).* The various indicators are described below.

3.10.1 Unbalanced Force

Forces are accumulated at each centroid of rigid blocks and each gridpoint of deformable blocks. At equilibrium – or steady plastic flow in deformable blocks – the algebraic sum of these forces is almost zero (i.e., the forces acting on one side of the block centroid or gridpoint nearly balance those acting on the other). During timestepping, the maximum unbalanced force is determined for the whole model; this force is displayed continuously on the screen. It can also be saved as a history and viewed as a graph. The unbalanced force is important in assessing the state of the model for static analysis, but its magnitude must be compared with the magnitude of typical internal forces acting in the model. In other words, it is necessary to know what constitutes a "small" force. A representative internal gridpoint force for deformable blocks may be found by multiplying stress by zone length perpendicular to the force, using values that are typical in the area of interest in the model. Denoting R as the ratio of maximum unbalanced force to the representative internal force, expressed as a percentage, the value of R will never decrease to zero. However, a value of 1% or 0.1% may be acceptable as denoting equilibrium, depending on the degree of precision required (e.g., $R = 1\%$ may be good enough for an intermediate stage in a sequence of operations, while $R = 0.1\%$ may be used if a final stress or displacement distribution is required for inclusion in a report or paper). Note that a low value of R only indicates that forces balance at all gridpoints. However, steady plastic flow may be occurring, without acceleration. In order to distinguish between this condition and "true" equilibrium, other indicators, such as those described below, should be consulted.

3.10.2 Block/Gridpoint Velocities

The velocities of rigid blocks and gridpoints of deformable blocks may be assessed by plotting the whole field of velocities, or by selecting certain key points in the model and tracking their velocities with histories (**block gridpoint history vel-x** or **block gridpoint history vel-y**). Both types of plots are useful. Steady-state conditions are indicated if the velocity histories show horizontal traces in their final stages. If they have all converged to near zero (in comparison to their starting values), then absolute equilibrium has occurred. If a history has converged to a nonzero value, then either the block is falling, or steady plastic flow is occurring at the block/gridpoint corresponding to that history. If one or more velocity history plots show fluctuating velocities, then the system is likely to be in a transient condition.

---

* Interpretation of the state of a model for a dynamic analysis is discussed in Section 4 in **Special Features**.

The plot of the field of velocity vectors is more difficult to interpret, since both the magnitudes and the nature of the pattern are important. As with gridpoint forces, velocities never decrease precisely to zero. The magnitude of velocity should be viewed in relation to the displacement that would occur if a significant number of steps (e.g., 1000) were to be executed. For example, if current displacements in the system are of the order of 1 cm, the maximum velocity in the velocity plot is 10^{-3} m/sec and the timestep is 10^{-5} sec, then 1000 steps would produce an additional displacement of 10^{-5} m, or 10^{-3} cm, which is 0.1% of the current displacements. In this case, it can be said that the system is in equilibrium, even if the velocities all seem to be "flowing" in one direction. More often, the vectors appear to be random (or almost random) in direction and (possibly) in magnitude. This condition occurs when the changes in gridpoint force fall below the accuracy limit of the computer, which is around six decimal digits. A random velocity field of low amplitude is an infallible indicator of block stability and no plastic flow.

If the vectors in the velocity field are coherent (i.e., there is some systematic pattern) and their magnitude is quite large (using the criterion described above), then either blocks are falling or slipping, plastic flow is occurring within blocks, or the system is still adjusting elastically (e.g., damped elastic oscillation is taking place). To confirm that continuing plastic flow is occurring, a plot of plasticity indicators should be consulted, as described below. If, however, the motion involves elastic oscillation, then the magnitude should be observed in order to indicate whether such movement is significant. Seemingly meaningful patterns of oscillation may be seen. However, if amplitude is low, then the motion has no physical significance.

3.10.3 Plastic Indicators for Block Failure

For most of the nonlinear block models in *UDEC*, the state attribute on the block plot displays those zones in which the stresses satisfy the yield criterion. Such an indication usually denotes that plastic flow is occurring, but it is possible for a block zone simply to "sit" on the yield surface without any significant flow taking place. It is important to look at the whole pattern of plasticity indicators to see whether a *mechanism* has developed. A failure mechanism is indicated if there is a contiguous line of active plastic zones that join two surfaces. The diagnosis is confirmed if the velocity plot also indicates motion corresponding to the same mechanism. Note that initial plastic flow often occurs at the beginning of a simulation, but subsequent stress redistribution unloads the yielding elements so that their stresses no longer satisfy the yield criterion ("yielded in past"). Only the actively yielding elements ("at yield surface") are important to the detection of a failure mechanism. If there is no contiguous line or band of active plastic zones between boundaries, two patterns should be compared before and after the execution of, say, 500 steps. Is the region of active yield increasing or decreasing? If it is decreasing, then the system is probably heading for equilibrium; if it is increasing, then ultimate failure *may* be possible.

If a condition of continuing plastic flow has been diagnosed, one additional question should be asked: Does the active flow band(s) include zones adjacent to artificial boundaries? The term "artificial boundary" refers to a boundary that does not correspond to a physical entity, but exists simply to limit the size of the model that is used. If plastic flow occurs along such a boundary, then the solution is not realistic, because the mechanism of failure is influenced by a nonphysical entity. This comment only applies to the final steady-state solution; intermediate stages may exhibit flow along boundaries.

3.10.4 Histories

In any problem, there are certain variables that are of particular interest (e.g., displacements may be of concern in one problem, but stresses may be of concern in another). Liberal use should be made of histories to track these important variables in the regions of interest. After some timestepping has taken place, the plots of these histories often provide the way to find out what the system is doing.

3.10.5 Factor of Safety

3.10.5.1 Strength Reduction Method

A "factor of safety" can be determined in *UDEC* for any selected parameter by calculating the ratio of the selected parameter value under given conditions to the parameter value that results in failure. For example:

$$F_L = \frac{\text{applied load to cause failure}}{\text{design load}}$$

$$F_\phi = \frac{\tan(\text{actual friction angle})}{\tan(\text{friction angle at failure})}$$

Note that the larger value is always divided by the smaller value (assuming that the system does not fail under the actual conditions). *The definition of failure must be established by the user.* A comparison of this approach based on strength reduction for determining factor of safety to that based upon limit analysis solutions is given by Dawson and Roth (1999) and Dawson et al. (1999). The strength-reduction method as applied in *UDEC* to calculate factor of safety is discussed in detail in Section 2 in **Theory and Background**.

The strength reduction method for determining factor of safety is implemented in *UDEC* through the **block factor-of-safety** command. This command implements an automatic search for factor of safety using the bracketing approach, as described in Dawson et al. (1999). The **block factor-of-safety** command may be given at any stage in a *UDEC* run, and can be applied to any block plasticity model in *UDEC*. During the solution process, strength properties will be changed, but the original state will be restored on completion, or if the **fos** search is terminated prematurely with the <Esc> key.

The strength reduction method has been tested in *UDEC* for application with the Mohr-Coulomb, ubiquitous-joint and modified Hoek-Brown model. (See Section 2 in **Theory and Background**.)

When *UDEC* is executing the **block factor-of-safety** command, the bracketing values for F are printed continuously to the screen so that you can tell how the solution is progressing. If terminated by the <Esc> key, the solution process terminates, and the best current estimate of **fos** is displayed. A save file that corresponds to the last nonequilibrium state is produced, so that velocity vectors, and so on, can be plotted. This allows a visualization of the failure mode.

'503X1051' is a simple slope model containing construction joints; it illustrates how the command works. The run stops at $F = 1.04$. Figure 3.51 plots shear strain as well as velocity, which allows the user to see that failure would be in the continuum rather than on the joint surfaces.

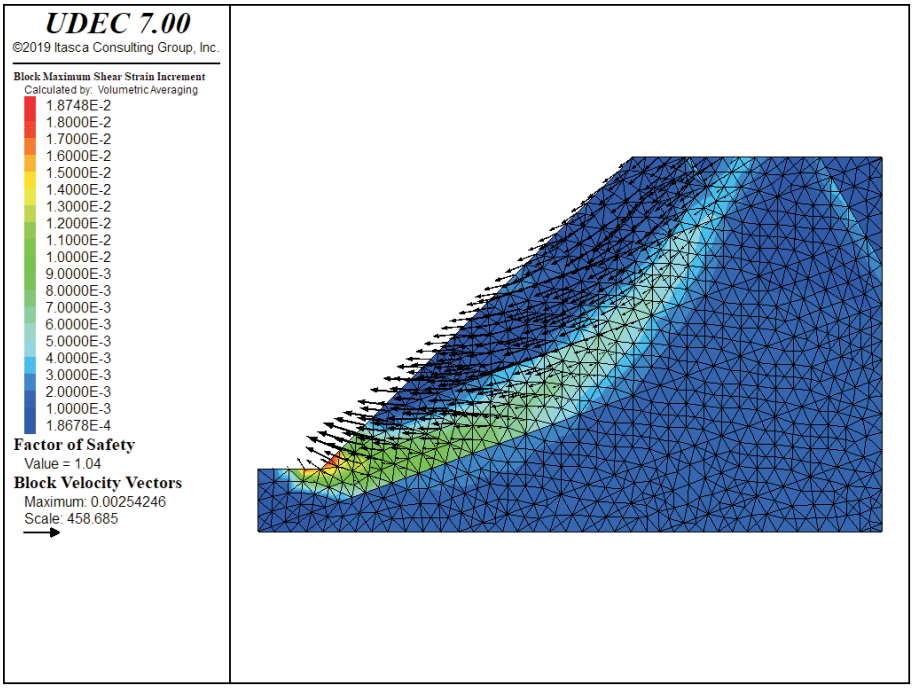

Figure 3.51 Shear strain contours and velocity vectors in slope model indicating location of failure surface

It is recommended that the **block factor-of-safety** command be given at an equilibrium state of a model (to improve solution time), but this is not essential.

The procedure used by *UDEC* during execution of **block factor-of-safety** is as follows. First the code finds a "representative number of steps" (denoted by N_r), which characterizes the response time of the system. N_r is found by setting the cohesion to a large value, making a large change to the internal stresses, and finding how many steps are necessary for the system to return to equilibrium. Then for a given factor of safety, F, N_r steps are executed. If the unbalanced force ratio is less than 10^{-5}, then the system is in equilibrium. If the unbalanced force ratio is greater than 10^{-5}, then another N_r steps are executed, exiting the loop if the force ratio is less than 10^{-5}. The mean value of force ratio, averaged over the current span of N_r steps, is compared with the mean force ratio over the previous N_r steps. If the difference is less than 10%, the system is deemed to be in nonequilibrium, and the loop is exited with the new nonequilibrium F. If the above-mentioned difference is greater than 10%, blocks of N_r steps are continued until (1) the difference is less than 10%, or (2) 6 such blocks have been executed, or (3) the force ratio is less than 10^{-5}. The justification for case (1) is that the mean force ratio is converging to a steady value that is greater than that corresponding to equilibrium; the system must be in continuous motion.

If a Mohr-Coulomb material model is assigned to blocks and Coulomb slip model to joints, then the default settings of the **block factor-of-safety** command will adjust the strength parameters of friction and cohesion for Mohr-Coulomb type failure in the zones and Coulomb slip in the joints. If the joint constitutive model does not use a Coulomb type failure, this process will not apply (i.e., the CY and BB joint models will not produce a factor of safety using this command).

There is an option to include other strength properties, such as tensile strength, in the calculations for either the zones or the joints. The strength reduction technique can also be selected for structural elements. In this case, these parameters are selected:

cables – grout shear strength, axial compressive and tensile yield strength

local reinforcement – axial and shear ultimate load limits

supports – axial compressive load limit

structural elements – axial compressive and tensile yield strengths

The user may also exclude certain material types from inclusion in the fos calculations. Section 2.4.1 in **Theory and Background** describes the inclusion and exclusion of strength properties for various models in more detail, and includes example applications.

3.10.5.2 Strength/Stress Ratio

It is also possible to plot a strength factor of safety which is a *strength/stress ratio* for zones in deformable blocks. These plots are independent of the **block factor-of-safety** command. Two strength criteria are available: Mohr-Coulomb and Hoek-Brown. The block plot contour attribute "mohr coulomb failure" plots contours of strength/stress ratio based on the Mohr-Coulomb criterion. And the block plot contout attribute "hoek brown failure" plots contours based on the Hoek-Brown criterion. The Mohr-Coulomb criterion is given in Section 1.6.2 in **Constitutive Models**. The Hoek-Brown criterion is described in Section 1.6.8 in **Constitutive Models**.

The state of stress within any zone can be expressed in terms of principal stresses σ_1 and σ_3. This stress state, in general, will plot as a circle, “a,” with a radius r_a, on the Mohr diagram (see Figure 3.52). Failure occurs if this circle just touches the failure envelope. The strength for the stress state represented by circle “a” is determined by holding σ_3 constant while increasing or decreasing σ_1 until circle “b,” with radius r_b, touches the envelope. The ratio of the radii of the two circles ($F = r_b/r_a$) is the strength/stress ratio. F is also known as the “failure index” or the “factor of safety.” Note that $|F| < 1$ for all circles “a” with points lying outside the envelope. F is zero whenever σ_3 is greater that the tension limit. For the Mohr-Coulomb criterion, this is taken as the smaller of the tensile strength and cohesion/tan (friction). For the Hoek-Brown criterion, the limit is taken as $s\ \sigma_c/m$ (as defined for Eq. (3.26)).

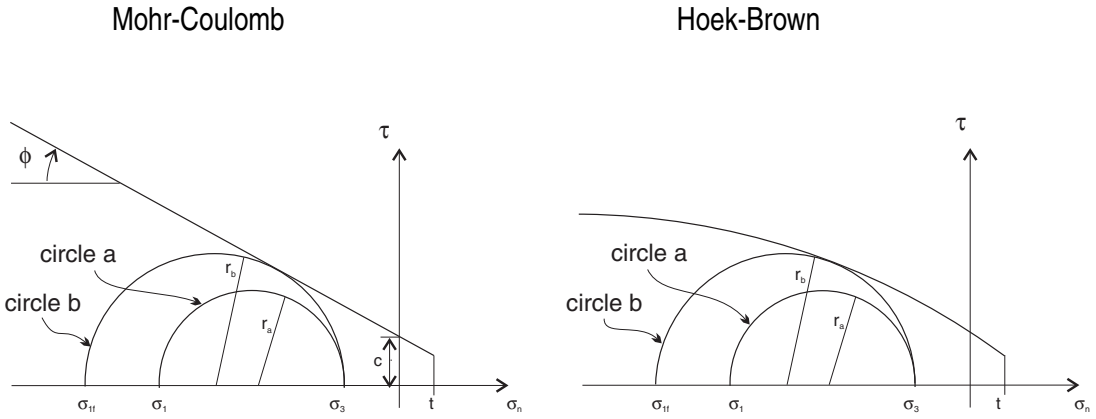

Figure 3.52 Strength/stress ratios for the Mohr-Coulomb and Hoek-Brown failure criteria

Eq. (3.39) shows the Mohr-Coulomb criterion while Eq. (3.40) shows the Hoek-Brown criterion. Eq. (3.41) is the strength/stress ratio.

$$\sigma_{1f} = \left(\frac{1+\sin\phi}{1-\sin\phi}\right)\sigma_3 - 2c\sqrt{\frac{1+\sin\phi}{1-\sin\phi}} \tag{3.39}$$

$$\sigma_{1f} = \sigma_3 - \sqrt{-m\ \sigma_c\sigma_3 + s\ \sigma_c^2} \tag{3.40}$$

$$r_b/r_a = \frac{\sigma_3 - \sigma_{1f}}{\sigma_3 - \sigma_1} \tag{3.41}$$

3.11 Modeling Methodology

3.11.1 Modeling of Data-Limited Systems

In a field such as geomechanics, where data are not always available, the methodology used in numerical modeling should be different from that used in a field such as mechanical engineering. Starfield and Cundall (1988) provide suggestions for an approach to modeling that is appropriate for a data-limited system. This paper should be consulted before any serious modeling with *UDEC* is attempted. In essence, the approach recognizes that field data (such as in-situ stresses, material properties and geological features) will never be known completely. It is futile to expect the model to provide design data, such as expected displacements, when there is massive uncertainty in the input data. However, a numerical model is still useful in providing a picture of the *mechanisms* that may occur in particular physical systems. The model acts to educate the intuition of the design engineer by providing a series of cause-and-effect examples. The models may be simple, with assumed data that are consistent with known field data and engineering judgement. It is a waste of effort to construct a very large and complicated model that may be just as difficult to understand as the real case.

Of course, if extensive field data are available, then these may be incorporated into a comprehensive model that can yield design information directly. More commonly, however, the data-limited model does not produce such information directly, but provides insight into mechanisms that may occur. The designer can then do simple calculations, based on these mechanisms, that estimate the parameters of interest or the stability conditions.

3.11.2 Modeling of Chaotic Systems

In some calculations, especially in those involving discontinuous materials, the results can be extremely sensitive to very small changes in initial conditions or trivial changes in loading sequence. Initially, this situation may seem unsatisfactory and may be taken as a reason to mistrust the computer simulations. However, the sensitivity exists in the physical system being modeled. There appear to be a least two sources for the seemingly erratic behavior:

1. There are certain geometric patterns of discontinuities that force the system to choose, apparently at random, between two alternative outcomes; the subsequent evolution depends on which choice is made. For example, Figure 3.53 illustrates a small portion of a jointed rock mass. If block A is forced to move down relative to B, it can either go to the left or to the right of B; the choice will depend on microscopic irregularities in geometry, properties or kinetic energy.

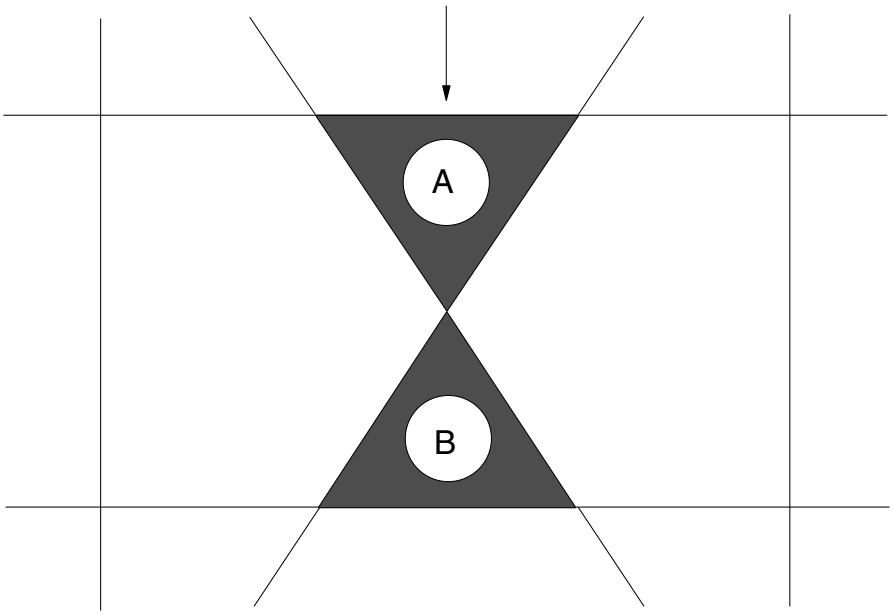

Figure 3.53 A small portion of a jointed rock mass

2. There are processes in the system that can be described as "softening" or, more generally, as cases of positive feedback. In a fairly uniform stress field, small perturbations are magnified in the subsequent evolution because a region that has more strain, softens more and thereby attracts more strain, and so on, in a cycle of positive feedback.

Both phenomena give rise to behavior that is chaotic in its extreme form (Gleick 1987, and Thompson and Stewart 1986). The study of chaotic systems reveals that the detailed evolution of such a system is not predictable, *even in principle*. The observed sensitivity of the computer model to small changes in initial conditions or numerical factors is simply a reflection of a similar sensitivity in the real world to small irregularities. There is no point in pursuing ever more "accurate" calculations, because the resulting model is unrepresentative of the real world, where conditions are not perfect. What should our modeling strategy be in the face of a chaotic system? It appears that the best we can expect from such a model is a finite *spectrum* of expected behavior; the *statistics* of a chaotic system are well-defined. We need to construct models that contain distributions of initial irregularities (e.g., by using the **block cut joint-set** command in *UDEC* with the random deviation parameter for joint characteristics). Each model should be run several times, with different distributions of joints. Under these conditions, we may expect the fluctuations in behavior to be triggered by the imposed irregularities, rather than by artifacts of the numerical solution scheme. We can express the results in a statistical form.

3.11.3 Localization, Physical Instability and Path-Dependence

In many systems that can be modeled with *UDEC*, there may be several paths the solution may take, depending on rather small changes in initial conditions. This phenomenon is termed *bifurcation*. For example, a shear test on an elastic/plastic material may either deform uniformly, or it may exhibit shear bands, in which the shear strain is localized rather than being uniformly distributed. It appears that if a numerical model has enough degrees of freedom (i.e., enough elements), then localization is to be expected. Indeed, theoretical work on the bifurcation process (e.g., Rudnicki and Rice 1975, and Vardoulakis 1980) shows that shear bands form even if the material does not strain-soften, provided that the dilation angle is lower than the friction angle. The "simple" Mohr-Coulomb material should always exhibit localization if there are enough elements to resolve one or more localized bands. A strain-softening material is more prone to produce bands.

Some computer programs appear incapable of reproducing band formation, although the phenomenon is to be expected physically. However, *UDEC* is able to allow bands to develop and evolve, partly because it models the dynamic equations of motion (i.e., the kinetic energy that accompanies band formation is released and dissipated in a physically realistic way). Several papers document the use of *FLAC* in modeling shear band formation (e.g., Cundall 1989, 1990 and 1991). These should be consulted for details concerning the solution process. One aspect that is not treated well by *UDEC* is the *thickness* of a shear band. In reality, the thickness of a band is determined by internal features of the material, such as grain size.

These features are not built into *UDEC*'s constitutive models. Hence, the bands in *UDEC* collapse down to the smallest width that can be resolved by the grid, which is one grid-width if the band is parallel to the grid, or about three grid-widths if the band cuts across the grid at an arbitrary angle. Although the overall physics of band formation are modeled correctly by *UDEC*, band thickness and band spacing are grid-dependent. Furthermore, if the strain-softening model is used with a weakening material, the load/displacement relation generated by *UDEC* for a simulated test is strongly grid-dependent. This is because the strain concentrated in a band depends on the width of the band (in length units), which depends on zone size, as we have seen. Hence, smaller zones lead to more softening, since we move out more rapidly on the strain axis of the given softening curve. To correct this grid dependence, some sort of length scale must be built into the constitutive model. There is controversy, at present, concerning the best way to do this. It is anticipated that future versions of *UDEC* will include a length scale in the constitutive models – probably involving the use of a Cosserat material, in which internal spins and moments are taken into account. In the meantime, the processes of softening and localization may be modeled, but it must be recognized that the grid size and angle affect the results; models must be calibrated for each grid used.

One topic that involves chaos, physical instability and bifurcation is *path-dependence*. In most nonlinear, inelastic systems, there are an *infinite* number of solutions that satisfy equilibrium, compatibility and the constitutive relations. There is no "correct" solution to the physical problem unless the path is specified. If the path is not specified, all possible solutions are correct. This situation can cause endless debate among modelers and users, particularly if a seemingly irrelevant parameter in the solution process (e.g., damping) is seen to affect the final result. All of the solutions are valid numerically. For example, a simulation done of a mining excavation with low damping may show a large overshoot and, hence, large final displacements, while high damping will eliminate the overshoot and give lower final displacements. Which one is more realistic? It depends on the

path. If the excavation is done by explosion (i.e., suddenly), then the solution with overshoot may be the appropriate one; if the excavation is done by pick and shovel (i.e., gradually), then the second case may be more appropriate. For cases in which path-dependence is a factor, modeling should be done in a way that mimics the way the system evolves physically.

3.11.4 Multithreading

The evolution of hardware in recent years has focused not on building faster processors, but on adding more computation cores.

In order to take advantage of multiple cores, the main calculation loops in *UDEC* have been modified to run on multiple threads. The multithreading has been implemented in the following calculations:

a. zone stress/strain calculations

b. contact force/displacement calculations

c. fluid-flow calculations

The multithreading in the stress/strain calculation is organized based on the blocks. All of the zones in each of the blocks are assigned a thread. This optimizes the calculation for models that have many blocks and many zones. A model with only a single block containing many zones will not execute in multiple threads. The multithreading for the contacts is organized based on the individual contacts. The calculations for each contact will occur on a single thread. The fluid-flow logic contains loops for domains and contacts. This is organized based on each domain, and on each contact. So the calculations for each domain and each contact will occur in a separate thread.

The efficiency that can be obtained through multithreading is not 100%. Many calculations are outside the multi-thread loops, and there is overhead in setting up the threads. Also, there are more cores in the CPUs than memory data paths. This can result in the CPUs becoming memory-bound as they wait to access memory. Currently, the speed increase in a *UDEC* 7.0 (compared to non mulyithreaded) model with many blocks and zones can be expected to be 2-3 times faster on a 4 core CPU with hyper-threading (8 computational units). The speed increase will vary depending on what the model is currently spending the most time doing.

The calculations in the multithreading have been designed to be deterministic. This means that you will get the same result each time the same data file is executed. This feature can be turned off with **SET deterministic off**. Turning deterministic off will result in a speed increase of approximately 10%.

3.12 References

Bandis, S. C., A. C. Lumsden and N. R. Barton. "Fundamentals of Rock Joint Deformation," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***20***(6), 249-268 (1983).

Barton, N. "The Shear Strength of Rock and Rock Joints," *Int. J. Rock Mech. Min. Sci. & Geotech. Abstr.*, ***13***, 255-279 (1976).

Bieniawski, Z. T. "Determining Rock Mass Deformability: Experience from Case Histories," *Int. J. Rock Mech. Min. Sci.*, ***15***, 237-247 (1978).

Board, M., et al. "Comparative Analysis of Toppling Behaviour at Chuquicamata Open-Pit Mine, Chile," *Trans. Instn. Min. Metall., Sec. A*, ***105***, A11-A21 (1996).

Brady, B. H. G., and E. T. Brown. ***Rock Mechanics for Underground Mining***. London: George Allen and Unwin. (1985).

Brady, B. H. G., J. V. Lemos and P. A. Cundall. "Stress Measurement Schemes for Jointed and Fractured Rock," in ***Rock Stress and Rock Stress Measurements***, pp. 167-176. Luleå, Sweden: Centek Publishers (1986).

Clark, I. H. "The Cap Model for Stress Path Analysis of Mine Backfill Compaction Processes," in ***Computer Methods and Advances in Geomechanics***, Vol. 2, pp. 1293-1298. Rotterdam: A. A. Balkema (1991).

Cundall, P. A. "Numerical Experiments on Localization in Frictional Material," *Ingenieur-Archiv*, ***59***, 148-159 (1989).

Cundall, P. A. "Numerical Modelling of Jointed and Faulted Rock," in ***Mechanics of Jointed and Faulted Rock***, pp. 11-18. Rotterdam: A. A. Balkema (1990).

Cundall, P. A. "Shear Band Initiation and Evolution in Frictional Materials," in ***Mechanics Computing in 1990s and Beyond (Proceedings of the Conference, Columbus, Ohio, May 1991)***, Vol. 2: Structural and Material Mechanics, pp. 1279-1289. H. Adeli and R. Sierakowski, eds. New York: ASME (1991).

Dawson, E. M., and W. H. Roth. "Slope Stability Analysis with FLAC," in ***FLAC and Numerical Modeling in Geomechanics (Proceedings of the International FLAC Symposium, Minneapolis, Minnesota, September 1999)***, pp. 3-9. C. Detournay and R. Hart, eds. Rotterdam: A. A. Balkema (1999).

Dawson, E. M., W. H. Roth and A. Drescher. "Slope Stability Analysis by Strength Reduction," *Géotechnique*, ***49***(6), 835-840 (1999).

Fossum, A. F. "Technical Note: Effective Elastic Properties for a Randomly Jointed Rock Mass," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***22***(6), 467-470 (1985).

Gleick, J. ***Chaos: Making a New Science***. New York: Penguin Books (1987).

Gerrard, C. M. "Equivalent Elastic Moduli of a Rock Mass Consisting of Orthorhombic Layers," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***19***, 9-14 (1982a).

Gerrard, C. M. "Elastic Models of Rock Masses Having One, Two and Three Sets of Joints," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***19***, 15-23 (1982b).

Goodman, R. E. ***Introduction to Rock Mechanics***. New York: John Wiley and Sons (1980).

Hoek, E. "Estimating Mohr-Coulomb Friction and Cohesion Values from the Hoek-Brown Failure Criterion," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***27***(3), 227-229 (1990).

Hoek, E., and E. T. Brown. "The Hoek-Brown Failure Criterion – a 1988 Update," in ***Rock Engineering for Underground Excavations***, pp. 31-38. Toronto: University of Toronto (1988).

Hoek, E., and E. T. Brown. ***Underground Excavations in Rock***. London: Instn. Min. Metall. (1980).

Huang, X., et al. "An Investigation of the Mechanics of Rock Joints – Part I. Laboratory Investigation," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***30***, 257-269 (1993).

Jaeger, J. C., and N. G. W. Cook. ***Fundamentals of Rock Mechanics***, 2nd Ed. London: Chapman and Hall (1969).

Jing, L., E. Nordlund and O. Stephansson. "An Experimental Study on the Anisotropy an Stress-Dependency of the Strength and Deformability of Rock Joints," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***29***, 535-542 (1992).

Kulhawy, Fred H. "Stress Deformation Properties of Rock and Rock Discontinuities," *Engineering Geology*, ***9***, 327-350 (1975).

Lorig, L. J., and B. H. G. Brady. "An Improved Procedure for Excavation Design in Stratified Rock," in ***Rock Mechanics – Theory-Experiment-Practice***, pp. 577-586. New York: Association of Engineering Geologists (1983).

Rosso, R. S. "A Comparison of Joint Stiffness Measurements in Direct Shear, Triaxial Compression, and In-Situ," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***13***, 167-172 (1976).

Rudnicki, J. W., and J. R. Rice. "Conditions for the Localization of the Deformation in Pressure-Sensitive Dilatant Materials," *J. Mech. Phys. Solids*, ***23***, 371-394 (1975).

Serafim, J. L., and J. P. Pereira. "Considerations of the Geomechanical Classification of Bieniawski," ***Proceedings of the International Symposium on Engineering Geology and Underground Construction (Lisbon, Portugal, 1983)***, Vol. 1, pp. II.33-42. Lisbon: SPGILNEC (1983).

Singh, B. "Continuum Characterization of Jointed Rock Masses: Part I – The Constitutive Equations," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***10***, 311-335 (1973).

Souley, M., F. Homand and B. Amadei. "An Extension to the Saeb and Amadei Constitutive Model for Rock Joints to Include Cyclic Load Paths," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***32***, 101-109 (1995).

Starfield, A. M., and P. A. Cundall. "Towards a Methodology for Rock Mechanics Modelling," *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, ***25***(3), 99-106 (1988).

Thompson, J. M. T., and H. B. Stewart. ***Nonlinear Dynamics and Chaos***. New York: John Wiley and Sons (1986).

Vardoulakis, I. "Shear Band Inclination and Shear Modulus of Sand in Biaxial Tests," *Int. J. Numer. Anal. Meth. in Geomechanics*, ***4***, 103-119 (1980).

Vermeer, P. A., and R. de Borst. "Non-Associated Plasticity for Soils, Concrete and Rock," *Heron*, ***29***(3), 3-64 (1984).